IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	:	
Lan BO et al.	:	
Serial No.	:	Art Unit:
Filed: November 4, 2003	:	Examiner:
For: AN MPEG-4 LIVE UNICAST VIDEO STREAMING SYSTEM IN WIRELESS NETWORK WITH END-TO-END BITRATE-BASED CONGESTION CONTROL	:	Atty Docket: 0124/0016

SUBMISSION OF PRIORITY DOCUMENT

Assistant Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Attached hereto please find a certified copy of applicants' Singapore patent application No. 200207018-3 filed November 20, 2002.

Applicants request the benefit of said November 20, 2003 filing date for priority purposes pursuant to the provisions of 35 USC 119.

Respectfully submitted,

Louis Woo, Reg. No. 31,730
Law Offices of Louis Woo
717 North Fayette Street
Alexandria, Virginia 22314
Phone: (703) 299-4090

Date: Nov 4 2003

REGISTRY OF PATENTS
SINGAPORE

This is to certify that the annexed is a true copy of specification as filed for the following Singapore patent application.

Date of Filing	:	20 NOVEMBER 2002
Application Number	:	200207018-3
Applicant(s) / Proprietor(s) of Patent	:	VICTOR COMPANY OF JAPAN, LTD.
Title of Invention	:	AN MPEG-4 LIVE UNICAST VIDEO STREAMING SYSTEM IN WIRELESS NETWORK WITH END-TO-END BITRATE-BASED CONGESTION CONTROL

SHARMAINE WU (Ms)
Assistant Registrar
for REGISTRAR OF PATENTS

PATENTS FORM 1
Patents Act
(Cap. 221)
Patents Rules
Rule 19

INTELLECTUAL PROPERTY OFFICE OF SINGAPORE

101101

REQUEST FOR THE GRANT OF A PATENT UNDER SECTION 25

* denotes mandatory fields

1. YOUR REFERENCE* 1013491PAT/Victor/ELK/Ls

2. TITLE OF INVENTION* AN MPEG-4 LIVE UNICAST VIDEO STREAMING SYSTEM IN WIRELESS NETWORK WITH END-TO-END BITRATE –BASED CONGESTION CONTROL

3. DETAILS OF APPLICANT(S)* (see note 3) Number of applicant(s) 1

(A) Name: VICTOR COMPANY OF JAPAN, LTD.

Address: 12, 3-chome, Moriya-cho, Kanagawa-ku,
Yokohama-shi, Kanagawa-ken 221-8528,
Japan

State: Country: JP

[X] For corporate applicant

State of incorporation:

Country of incorporation: JP

[ ] For individual applicant

State of residency:

Country of residency:

[ ] For others (please specify in the box provided below)

(B) Name:

Address:

State: Country:

20 NOV 2002
200207018-3

ACTION

☐ For corporate applicant

State of incorporation

Country of incorporation

☐ For individual applicant

State of residency

Country of residency

☐ For others (please specify in the box provided below)

(C) Name

Address

State

Country

☐ For corporate applicant

State of incorporation

Country of incorporation

☐ For individual applicant

State of residency

Country of residency

☐ For others (please specify in the box provided below)

☐ Further applicants are to be indicated on continuation sheet 1

---

4. DECLARATION OF PRIORITY (see note 5)

A. Country/country designated

File number

Filing Date (DD MM YYYY)

B. Country/country designated

File number

Filing Date (DD MM YYYY)

☐ Further details are to be indicated on continuation sheet 6

---

5. INVENTOR(S)* (see note 6)

A. The applicant(s) is/are the sole/joint inventor(s) Yes ☐ No ☒

20 NOV 2002

200207018-3

B. A statement on Patents Form 8 is/will be furnished Yes [X] No [ ]

6. CLAIMING AN EARLIER FILING DATE UNDER (see note 7)

[ ] section 20(3) [ ] section 26(6) [ ] section 47(4)

Patent application number [ ]

Filing Date (DD MM YYYY) [ ]

Please mark with a cross in the relevant checkbox provided below
(Note: Only one checkbox may be crossed.)

[ ] Proceedings under rule 27(1)(a)

Date on which the earlier application was amended (DD MM YYYY) [ ]

[ ] Proceedings under rule 27(1)(b)

7. SECTION 14(4)(C) REQUIREMENTS (see note 8)

Invention has been displayed at an international exhibition. Yes [ ] No [X]

8. SECTION 114 REQUIREMENTS (see note 9)

The invention relates to and/or used a micro-organism deposited for the purposes of disclosure in accordance with section 114 with a depository authority under the Budapest Treaty.

Yes [ ] No [X]

9. CHECKLIST*

(A) The application consists of the following number of sheets

i.	Request	5	Sheets
ii.	Description	26	Sheets
iii.	Claim(s)	6	Sheets
iv.	Drawing(s)	18	Sheets
v.	Abstract (Note: The figure of the drawing, if any, should accompany the abstract)	1	Sheets
Total number of sheets		56	Sheets

(B) The application as filed is accompanied by:

[ ] Priority document(s) [ ] Translation of priority document(s)

☒	Statement of inventorship & right to grant	☐	International exhibition certificate

10. DETAILS OF AGENT (see notes 10, 11 and 12)

Name:

Firm: DONALDSON & BURKINSHAW

11. ADDRESS FOR SERVICE IN SINGAPORE* (see note 10)

Block/Hse No.: Level No.: Unit No./PO Box: 3667

Street Name:

Building Name:

Postal Code: 905667

12. NAME, SIGNATURE AND DECLARATION (WHERE APPROPRIATE) OF APPLICANT OR AGENT* (see note 12)
(Note: Please cross the box below where appropriate.)

☒ I, the undersigned, do hereby declare that I have been duly authorised to act as representative, for the purposes of this application, on behalf of the applicant(s) named in paragraph 3 herein.

Name and Signature
(DONALDSON & BURKINSHAW)

DD MM YYYY
20 11 2002

20 NOV 2002

200207018-3

NOTES:

1. This form when completed, should be brought or sent to the Registry of Patents together with the rest of the application. Please note that the filing fee should be furnished within the period prescribed.

2. The relevant checkboxes as indicated in bold should be marked with a cross where applicable.

3. Enter the name and address of each applicant in the spaces provided in paragraph 3.
Where the applicant is an individual
- Names of individuals should be indicated in full and the surname or family name should be underlined.
-The address of each individual should also be furnished in the space provided.
- The checkbox for "For individual applicant" should be marked with a cross.

Where the applicant is a body corporate
- Bodies corporate should be designated by their corporate name and country of incorporation and, where appropriate, the state of incorporation within that country should be entered where provided.
- The address of the body corporate should also be furnished in the space provided.
- The checkbox for "For corporate applicant" should be marked with a cross.

Where the applicant is a partnership
- The details of all partners must be provided. The name of each partner should be indicated in full and the surname or family name should be underlined.
- The address of each partner should also be furnished in the space provided.
- The checkbox for "For others" should be marked with a cross and the name and address of the partnership should be indicated in the box provided.

4. In the field for "Country", please refer to the standard list of country codes made available by the Registry of Patents and enter the country code corresponding to the country in question.

5. The declaration of priority in paragraph 4 should state the date of the previous filing, the country in which it was made, and indicate the file number, if available. Where the application relied upon in an International Application or a regional patent application e.g. European patent application, one of the countries designated in that application [being one falling under section 17 of the Patents Act] should be identified and the country should be entered in the space provided.

6. Where the applicant or applicants is/are the sole inventor or the joint inventors, paragraph 5 should be completed by marking with a cross the 'YES' checkbox in the declaration (A) and the 'NO' checkbox in the alternative statement (B). Where this is not the case, the 'NO' checkbox in declaration (A) should be marked with a cross and a statement will be required to be filed on Patents Form 8.

7. When an application is made by virtue of section 20(3), 26(6) or 47(4), the appropriate section should be identified in paragraph 6 and the number of the earlier application or any patent granted thereon identified. Applicants proceeding under section 26(6) should identify which provision in rule 27 they are proceeding under. If the applicants are proceeding under rule 27(1)(a), they should also indicate the date on which the earlier application was amended.

8. Where the applicant wishes an earlier disclosure of the invention by him at an International Exhibition to be disregarded in accordance with section 14(4)(c), then the 'YES' checkbox at paragraph 7 should be marked with a cross. Otherwise, the 'NO' checkbox should be marked with a cross.

9. Where in disclosing the invention the application refers to one or more micro-organisms deposited with a depository authority under the Budapest Treaty, then the 'YES' checkbox at paragraph 8 should be marked with a cross. Otherwise, the 'NO' checkbox should be marked with a cross. Attention is also drawn to the Fourth Schedule of the Patents Rules.

10. Where an agent is appointed, the fields for "DETAILS OF AGENT" and "ADDRESS FOR SERVICE IN SINGAPORE" should be completed and they should be the same as those found in the corresponding Patents Form 41. In the event where no agent is appointed, the field for "ADDRESS FOR SERVICE IN SINGAPORE" should be completed, leaving the field for "DETAILS OF AGENT" blank.

11. In the event where an individual is appointed as an agent, the sub-field "Name" under "DETAILS OF AGENT" must be completed by entering the full name of the individual. The sub-field "Firm" may be left blank. In the event where a partnership/body corporate is appointed as an agent, the sub-field "Firm" under "DETAILS OF AGENT" must be completed by entering the name of the partnership/body corporate. The sub-field "Name" may be left blank.

12. Attention is drawn to sections 104 and 105 of the Patents Act, rules 90 and 105 of the Patents Rules, and the Patents (Patent Agents) Rules 2001.

13. Applicants resident in Singapore are reminded that if the Registry of Patents considers that an application contains information the publication of which might be prejudicial to the defence of Singapore or the safety of the public, it may prohibit or restrict its publication or communication. Any person resident in Singapore and wishing to apply for patent protection in other countries must first obtain permission from the Singapore Registry of Patents unless they have already applied for a patent for the same invention in Singapore. In the latter case, no application should be made overseas until at least 2 months after the application has been filed in Singapore, and unless no directions had been issued under section 33 by the Registrar or such directions have been revoked. Attention is drawn to sections 33 and 34 of the Patents Act.

14. If the space provided in the patents form is not enough, the additional information should be entered in the relevant continuation sheet. Please note that the continuation sheets need not be filed with the Registry of Patents if they are not used.

20 NOV 2002

200207018-3

An MPEG-4 LIVE UNICAST VIDEO STREAMING SYSTEM IN WIRELESS NETWORK WITH END-TO-END BITRATE-BASED CONGESTION CONTROL

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to the realization of an MPEG-4 Live Unicast Video Streaming System in wireless network with end-to-end congestion control, which can automatically and dynamically adjust the data-bitrate/transmission-bitrate according to the available network bandwidth.

2. Description of the Related Art

Due to the success of the Internet technology and increasing demand for multimedia information on the web, streaming video over the Internet has becoming a key topic in academia and industry.

Conventionally, video files are downloaded from the Internet and played back locally. But full file transfer introduces very long, unacceptable transfer time and playback latency. No live streaming is supported.

Recently, with the emergence of wideband network, such as DSL (the Digital Subscriber Loop), cable modems, etc., real-time video streaming over the Internet is widely accepted and deployed. In real-time video streaming, live video or stored video is streamed across the Internet from the server to the client in response to a client's request. The client plays back the incoming video in real time when the data is received. There are several key areas of real-time video streaming such as video compression, application-layer QoS Quality of Service) control, continuous media distribution services, streaming servers, media synchronization mechanisms, protocols for streaming media, etc. Typically, real-time video streaming has bandwidth, delay, and loss requirements. For instance, video data must be played back continuously at the Client. If the data does not arrive in

time, the playback probably has to be paused to wait for the delayed or lost data, which is annoying to the user. However, the current best-effort Internet does not offer any QoS guarantees to streaming video. One solution is the application-layer QoS control, which does not require any QoS support from the network, to avoid congestion and maximize video quality in the presence of packet loss and transmission delay. For video streaming, typical application-layer congestion control takes the form of Rate Control, which attempts to match the bitrate of the video stream to the available network bandwidth.

One popular Rate Control is RAP (Rate Adaptation Protocol), which is end-to-end TCP-friendly. Video data is streamed through TCP connection. In the feedback TCP Acknowledgement, there is the information of RTT (Round-Trip Time), packet loss ratio, etc. Then an estimated current network bandwidth is achieved to adjust the sending data bitrate of a pre-stored video stream.

Another popular Rate Control is Receiver-Based Rate Control, which is mainly used in multicasting scalable video streams. There are several layers in the scalable video and each layer corresponds to one channel in the multicast tree. When congestion is detected, a receiver drops a layer resulting in a reduction of its receiving rate whereas the sender does not participate in Rate Control.

Now quite a few protocols have been designed and standardized for communication between clients and streaming servers. Obviously, IP serves as the network-layer protocol for the Internet video streaming. Since TCP's retransmission feature always introduces delays that are not acceptable for video streaming applications with stringent delay requirements, UDP is now typically employed as the transport protocol for video streaming. In addition, RTP/RTCP (Real-time Transport Protocol/Real-Time Control Protocol) is designed to provide end-to-end transport functions on top of UDP for supporting real-time applications. Moreover, RTSP (Real Time

Streaming Protocol) defines the messages and procedures to control the delivery of the multimedia data during an established session.

With the explosive development on wireless network, more and more people are now using wireless LAN or even their hand-phones and PDAs to access Internet. However, compared to the wired network, wireless links are more error-prone, bandwidth-limited and time varying. Thanks to the flexibility and efficiency of MPEG-4 technology, Video streaming through wireless network becomes available, especially for live video streaming.

SUMMARY OF THE INVENTION

In view of the foregoing, it is an object of the present invention to provide a network bandwidth/bitrate adaptation method, for use in an MPEG-4 Live Unicast Video Streaming system in wireless network, that allows the Streaming Server to provide continuous video streaming service over best-effort network. At the same time, it is another object of the present invention to provide a real-time streaming method including packetization, retransmission, bitrate adjustment, etc., for use with RTP/RTCP/UDP and RTSP, that allows the Client to receive data in real-time and decode data properly.

To solve the above problems, it is provided that the downlink (streaming channel) adopts UDP (User Datagram Protocol, RFC768) and the uplink (messaging channel) adopts TCP (Transmission Control Protocol, RFC793). The present invention provides the architecture of transporting MPEG-4 Simple Profile Video data, which is shown in FIG. 1, wherein,

(1) A Rate Adaptive MPEG-4 Simple Profile Encoder, which is out of the scope of this invention

- (a) Generates the MPEG-4 Simple Profile live video data,
- (b) Pushes live video data to the Streaming Server through LAN (Local Area Network),

(c) Adjusts the encoding bitrate in accordance with the request from the Streaming Server,

(2) A LAN (Local Area Network)

(a) Connects the Rate Adaptive MPEG-4 Simple Profile Encoder and the Streaming Server,

(3) A Streaming Server

(a) Has a Data Receiver module to receive live MPEG-4 video data from the Rate Adaptive MPEG-4 Simple Profile Encoder through LAN,

(b) Has a RTSP (Real Time Streaming Protocol, RFC2326) Server module, which performs the session control,

(c) Has a Data Transmission module (the RTP/RTCP Transport Engine Server), which segmentizes the MPEG-4 data on the boundary of GOV (Group of Video Object Plane), packetizes each GOV as the payload of RTP (Real Time Transport Protocol, RFC1889) packets, and pushes those RTP/UDP packets to the Client through the wireless network according to each GOV data bitrate, whereas RTCP (Real Time Control Protocol, RFC1889) is implemented to receive the retransmission request,

(d) Has a Bitrate Adapter module, which implements the Bitrate Adaptation protocol and Network Bandwidth Polling protocol, to allow the Streaming Server to receive feedback information from the Client and make the decision on bitrate control that is to be forwarded to the Encoder,

(e) Has a Data Link Buffer, which is to store the GOV data that is received from the Rate Adaptive MPEG-4 Simple Profile Encoder,

(4) A Rate Adaptive MPEG-4 Simple Profile Decoder, which is out of the scope of this invention

(a) Resides in the Client application,

(b) Decodes the received MPEG-4 Simple Profile live video data,

(c) Renders decoded pictures,

(5) A Client

(a) Receives live MPEG-4 video data from the Streaming Server through wireless network (Internet),

(b) Has a RTSP (Real Time Streaming Protocol, RFC2326) Client module, which performs the session control,

(c) Has a Data Transmission module (the RTP/RTCP Transport Engine Client), which receives the RTP/UDP packets from the Streaming Server through the wireless network (Internet), depacketizes each GOV from the payload of RTP (Real Time Transport Protocol, RFC1889) packets, desegmentizes the MPEG-4 data back to GOV (Group of Video Object Plane), and reconstructs the MPEG-4 video stream, whereas RTCP (Real Time Control Protocol, RFC1889) is implemented to send the retransmission request,

(d) Has a Bitrate Adapter module, which implements the Bitrate Adaptation protocol and Network Bandwidth Polling protocol, to feedback bitrate control information to the Streaming Server,

(e) Has a Data Link Buffer, which is to store the received GOV data and monitor the buffering status of itself, as well as forward the collected buffer state information to the Bitrate Adapter module.

According to the present invention, the initial streaming bitrate is to be decided by two ways, that is

(1) Manually configured at the Client by the user through a GUI, or

(2) Auto-negotiated by the Streaming Server and the Client with the Network Bandwidth Polling protocol.

Furthermore, in the present invention, the bitrate adaptation to the available network bandwidth consists of two aspects, that is

(1) Decrease of encoding bitrate due to network deterioration or decoder's poor throughput, and

(2) Increase of encoding bitrate due to health network condition.

In a preferred embodiment of the present invention, the Bit-stream, which is generated by the Rate Adaptive MPEG-4 Simple Profile Encoder, is firstly sent to the Streaming Server through HTTP/TCP connection with one GOV by one GOV.

The embodiment allows the Data Transmission module in the Streaming Server to segmentize and packetize the GOVs before the GOVs are put on the network according to their bitrate.

In a preferred embodiment of the present invention, if the Data Transmission module in the Client receives the incoming RTP/UDP packets, it starts the reconstruction of each GOV, in other words, the Access Unit. Then the recovered GOV is to be inserted into the Data Link Buffer in the Client.

In a preferred embodiment of the present invention, if packet loss occurs during the transmission of RTP/UDP packets, the blank GOV(s) or partially recovered GOV(s) (incomplete GOV(s)) is (are) to be inserted into the Data Link Buffer.

In a preferred embodiment of the present invention, it is the Data Link Buffer that checks whether it is necessary for a GOV or part of a GOV to be retransmitted. The retransmission checking is triggered by the insertion of a fully recovered GOV. Retransmission requests are to be passed to the Data Transmission module from the Data Link Buffer; then transmitted to the Streaming Server by the RTCP/UDP packets.

The embodiment allows the retransmission of a GOV or part of a GOV to be tried for only once.

The embodiment allows only those GOVs that are still in the Streaming Server's Data Link Buffer to be retransmitted.

In a preferred embodiment of the present invention, the Adaptive Rate MPEG-4 Simple Profile Decoder takes fully recovered GOVs, including those that are recovered by retransmission, from the Data Link Buffer.

In a preferred embodiment of the present invention, the Data Link Buffer in the Client collects its own current status and forwards that information to the Bitrate Adapter module in the Client. This is triggered each time when the Decoder successfully takes out a GOV from the Data Link Buffer.

In a preferred embodiment of the present invention, the Bitrate Adapter module in the Client evaluates the Bitrate Control Information; then forwards that information to its counterpart in the Streaming Server through TCP connection.

In a preferred embodiment of the present invention, the Bitrate Adapter module in the Streaming Server makes the decision on bitrate adjustment based on the information from its counterpart in the Client. Corresponding command is to be sent to the Adaptive Rate MPEG-4 Simple Profile Encoder to adjust the next GOV's encoding bitrate.

In a preferred embodiment of the present invention, the Bitrate Adapter module in the Client and its counterpart in the Streaming Server are to negotiate the initial streaming bitrate using Network Bandwidth Polling protocol by temporarily opening a UDP connection.

The embodiment allows the Network Bandwidth Polling process to be triggered by a Polling Timer during the data streaming procedure. The Bitrate Adapter module in the Client and its counterpart in the Streaming Server are to negotiate how far the current network bandwidth is over the current streaming bitrate by temporarily opening a UDP connection.

In a preferred embodiment of the present invention, a user controls the streaming session through GUI (Graphic User Interface). Commands such as Start, Stop, etc., are transported by RTSP/TCP connection.

The nature, principle and utility of the invention will become more apparent from the following detailed description when read in conjunction with the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

In the accompanying drawings:

FIG.1 is a diagram showing the system architecture of the present invention;

FIG.2 is a diagram showing the overview of the RTSP session procedure;

FIG.3 is a diagram showing the overview of the RTP/RTCP Transport Engine;

FIG.4 is a diagram showing the overview of the normal data transmission between the RTP/RTCP Transport Engines;

FIG.5 is a diagram showing the overview of the data retransmission between the RTP/RTCP Transport Engines;

FIG.6 is a diagram showing the structure of the Extended RTP Packet;

FIG.7 is a diagram showing the structure of the User Application RTCP Packet;

FIG.8 is an operation flowchart schematically showing the processing operation of a RTP/RTCP Transport Engine Server;

FIG.9 is a diagram showing the structure of a complete GOV with RG (RTP GOV);

FIG.10 is a diagram showing the classification of three different RGs regarding the UDP out-of-sequence problem.

FIG.11 is an operation diagram showing the processing operation of a RTP/RTCP Transport Engine Client;

FIG.12 is an operation diagram listing the main nine different processes of a RTP/RTCP Transport Engine Client upon receiving a RTP packet;

FIG.13 is a flowchart showing the process of GOV insertion in the Data Link Buffer (Client);

FIG.14 is a flowchart showing the process of GOV reading in the Data Link Buffer (Client);

FIG.15 is a time sequence diagram showing the Bitrate Control message flows in the current invention;

FIG.16 is a time sequence diagram showing the Retransmission message flows in the current invention;

FIG.17 is a diagram showing the process of making Bitrate Control Decision;

FIG.18 is an illustrative diagram showing the basic definitions of the Bitrate Control mechanism;

FIG.19 is an illustrative diagram showing the normal playback scenario of the Bitrate Control mechanism;

FIG.20 is an illustrative diagram showing the network deterioration scenario of the Bitrate Control mechanism;

FIG.21 is an illustrative diagram showing the Decoder's poor throughput scenario of the Bitrate Control mechanism;

FIG.22 is a time sequence diagram showing the polling process;

FIG.23 is a diagram showing the auto-negotiation procedure.

In the accompanying tables:

Table 1 explains all the fields in the Extended RTP packet;

Table 2 explains all the fields in the User Application RTCP packet.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

An embodiment of an MPEG-4 Live Unicast Video Streaming System in wireless network with end-to-end congestion control according to the present invention will be described in detail below with reference to the drawings.

A Streaming Server, a Client (including a Rate Adaptive MPEG-4 Simple Profile Decoder), and a Rate Adaptive MPEG-4 Simple Profile Encoder included in this embodiment use information-processing devices that can execute the process described below. Those information-processing devices include so-called general-purpose computers, workstations,

and personal computers, as well as network connectable information-processing devices such as digital home electric appliances, portable terminals such as PDAs, and cellular phones. It should be noted that the process described below may be performed by a software product and that a part of process may be done on a hardware unit.

FIG.1 is a diagram schematically showing the overview of a wireless MPEG-4 streaming system to which the MPEG-4 Live Unicast Video Streaming System according to the present invention is applied. Referring to the figure, the system comprises a Rate Adaptive MPEG-4 Simple Profile Encoder that encodes MPEG-4 data, a Streaming Server that receives MPEG-4 data from the Encoder and sends it to the Client, and a Client that receives MPEG-4 data and decodes it. The Encoder and the Streaming Server are connected via a LAN. The Client and the Streaming Server are connected via a wireless network.

FIG. 2 is a diagram schematically showing the overview of the session procedure of RTSP. The RTSP session procedure between the Streaming Server and the Client is fully conforming to RFC2326 with the following messages.

- Set up the session

Client ->Streaming Server
SETUP rtsp://*Server_Addr*:*Port_Num*/*Content_ID*/*User_ID* **RTSP/1.0**
Cseq: *Sequence_Num*
Transport: RTP/AVP; unicast; client_port=*RTP_Port-RTCP_Port*

Streaming Server -> Client
RTSP/1.0 200 OK
Cseq: *Sequence_Num*
Session: *Session_Num*
Transport: RTP/AVP;unicast;server_port=*RTP_Port-RTCP_Port*

- Start to play

Client ->Streaming Server
PLAY rtsp://*Server_Addr*:*Port_Num*/*Content_ID*/*User_ID* **RTSP/1.0**
Cseq: *Sequence_Num*
Session: *Session_Num*

Streaming Server -> Client
RTSP/1.0 200 OK
Cseq: *Sequence_Num*
Session: *Session_Num*
Range: npt=*Start_Time-End_Time*

- Stop the play and tear down the connection

Client -> Streaming Server
TEARDOWN rtsp://*Server_Addr:Port_Num/Content_ID/User_ID* **RTSP/1.0**
Cseq: *Sequence_Num*
Session: *Session_Num*

Streaming Server -> Client
RTSP/1.0 200 OK
Cseq: *Sequence_Num*
Session: *Session_Num*

- Keep alive message

Client -> Streaming Server
GET_PARAMETER rtsp://*Server_Addr:Port_Num/Content_ID/User_ID* **RTSP/1.0**
Cseq: *Sequence_Num*
Session: *Session_Num*

Streaming Server -> Client
RTSP/1.0 200 OK
Cseq: *Sequence_Num*
Session: *Session_Num*

The RTSP Client initially sends the SETUP message to the RTSP Server asking for a session setup. If the RTSP Client gets active ACK, it creates the object of RTP/RTCP Transport Engine Client and sends PLAY message to the RTSP Server asking for start of the streaming. The RTSP Server then instantiates the RTP/RTCP Transport Engine Server object to provide the streaming service. During the session, both of the RTSP Server and RTSP Client need to sense the counterpart's status by sending GET_PARAMETER messages to act as Keep Alive messages. At the end of the session, the RTSP Client sends the TEARDOWN message to terminate the session.

FIG. 3 is a diagram schematically showing the overview of the RTP/RTCP Transport Engine. The present invention requires the Data Link Buffer at both Client and Streaming Server sides to facilitate recovery of packet losses and ensure the constant flow of GOV data to the Decoder. At the Streaming Server, a GOV is firstly fragmented and encapsulated into RTP packets by the RTP/RTCP Transport Engine Server; then sent to the Client. At the Client, the RTP/RTCP Transport Engine Client extracts the GOV RTP data and assembles them into a GOV; then inserts it into the Client's Data Link Buffer. The followings summarize the main functionality of each component.

(1) Data Link Buffer (Server)
   a. Stores GOVs for transmission and retransmission,
(2) RTP/RTCP Transport Engine Server
   a. Handles transmission/retransmission of GOV data,
   b. Includes CRTP (Server), CRTCP (Server), Logger (Server), and Push Timer,
(3) CRTP (Server)
   a. Constructs the RTP Header from the related GOV information,
   b. Handles RTP/UDP connection,
   c. Handles packet transmission,
   d. Handles packet retransmission,
(4) CRTCP (Server)
   a. Receives retransmission request,
   b. Replies any retransmission-forbidden notice,
(5) Logger (Server)
   a. Records packet transmission timing,
   b. Logs error information,
(6) Push Timer
   a. Controls streaming bitrate,
(7) Data Link Buffer (Client)
   a. Stores GOVs that are from RTP/RTCP Transport Engine Client,

b. Passes GOVs to the Decoder,

(8) RTP/RTCP Transport Engine Client

a. Handles GOV data receiving,

b. Includes CRTP (Client), CRTCP (Client), and Logger (Client),

(9) CRTP (Client)

a. Handles RTP/UDP connection and data receiving,

b. Interprets RTP Header and extracts RTP payload data,

c. Reconstructs GOV,

(10) CRTCP (Client)

a. Sends out retransmission request,

b. Receives any retransmission-forbidden notice,

(11) Logger (Client)

a. Records RTP packet arrival timing,

b. Logs Error information.

FIG.4 is a diagram schematically showing the overview of the normal data transmission between the RTP/RTCP Transport Engines, wherein,

(1) At the Streaming Server side

a. Raw GOV (from the Encoder) is inserted into the Data Link Buffer (Server) with related information such as GOV bitrate, GOV duration, GOV size, etc,

b. A new memory node is allocated to store the newly inserted GOV with its related information,

c. RTP/RTCP Transport Engine Server takes one GOV node from the Data Link Buffer (Server),

d. RTP/RTCP Transport Engine Server calculates the RTP packet duration according to the GOV bitrate, GOV size, and RTP packet size,

e. RTP/RTCP Transport Engine Server sets the Push Timer,

f. When the Push Timer triggers, extended RTP packet (fragment of GOV) is sent to the Client,

(2) At the Client side

a. RTP/RTCP Transport Engine Client receives RTP packets containing GOV data,

b. RTP/RTCP Transport Engine Client try to reconstruct the GOV,

c. RTP/RTCP Transport Engine Client inserts the successfully recovered GOV into the Data Link Buffer (Client),

d. Data Link Buffer (Client) checks for any GOV that needs to be retransmitted wholly or partially and passes the request to the RTP/RTCP Transport Engine Client.

e. Retransmission requests are proceeded between the RTP/RTCP Transport Engines.

FIG.5 is a diagram schematically showing the overview of the data retransmission between the RTP/RTCP Transport Engines. In general, there are two types of retransmission, i.e., the retransmission of single RTP packet and the retransmission of a whole GOV. The corresponding requests are handled by CRTCP, which realizes RTCP protocol. In each retransmission request, the GOV sequence number and its RTP packet sequence number are defined as the fields of RTCP packet. Upon receiving such RTCP request, the Streaming Server will try to retrieve the designated GOV from the Data Link Buffer (Server) as long as the designated GOV is still there without being overwritten by new GOV. For request of whole GOV, the Streaming Server will try to push the designated GOV to the Client as soon as possible in multiple RTP packets. For request of single RTP packet, the Streaming Server takes out the corresponding chunk of data from the designated GOV according to the RTP sequence number. Then the Streaming Server pushes the corresponding chunk of data to the Client in one RTP packet as soon as possible. In the case of multiple RTP packets losses of one GOV, the retransmission requests of those RTP packets

will be issued one by one. However, if the designated GOV is no longer existing in the Data Link Buffer (Server), i.e., being overwritten by some new GOVs, the Streaming Server will reply a retransmission-forbidden message to the Client through a RTCP packet. Once the Client receives such a reply, it marks up a retransmission-forbidden flag of the corresponding GOV in the Data Link Buffer (Client) to indicate that retransmission is failed and no retransmission request should be re-issued. Retransmission RTP packet is processed as same as normal RTP packet is. The retransmitted data will be inserted into the corresponding GOV in the Data Link Buffer (Client) as long as it is still waiting for the Decoder's taking out. It is the Data Link Buffer (Client) that checks for any GOVs needing to be retransmitted. The process is triggered by the insertion of a new fully recovered GOV. Furthermore, in order not to affect the normal transmission too much, the number of retransmission request for the same GOV data should be limited, e.g., less than n times, in the case of retransmission packet being lost, since retransmission will consume the network bandwidth as well.

FIG.6 is a diagram showing the structure of the Extended RTP Packet. The RTP packet in the present invention is extended with several additional header fields to help packetize and depacketize the GOV. Table 1. explains all the fields in the Extended RTP packet, which is shown below,

Table 1. Fields in The Extended RTP Packet.

Field	***Size (bit)***	***Function***
V=2	2	As in RFC1889
P	1	As in RFC1889
X	1	As in RFC1889
CC	4	As in RFC1889
M	1	As in RFC1889
PT	7	As in RFC1889
Seq Num	8	As in RFC1889

Time Stamp	32	As in RFC1889
SSRC	32	As in RFC1889
Profile	16	Extended Profile
Length	16	Extended header length
TxGOVSeqNum	32	GOV Sequence Number used in Transmission
Tr	16	Total RTP packet for this GOV
Cr	16	Current RTP number for this GOV
SENTT	32	Server Estimated Next Transmission Time
dwGOVDataBuffSize	32	The byte's number of the GOV data buffer
dwGOVDataSize	32	The actual byte's number of GOV data
dwPreviousGOVBitrate	32	The bitrate of last GOV
dwCurrentGOVBitrate	32	The bitrate of current GOV
dwCurrentGOVPTS	32	The starting PTS of current GOV
dwCurrentGOVDuration	32	The duration of current GOV
dwCurrentGOVSeqNum	32	The unique sequence number of current GOV
DwNumOfFrame	32	The frame number of current GOV
dwCurrentGOVTransmissionSeqNum	32	The transmission sequence number of current GOV

FIG.7 is a diagram showing the structure of the User Application RTCP Packet. RTCP is used to send the retransmission requests for the lost RTP packets. Table 2 shows all the fields in the User Application RTCP packet as follows.

Table 2. The Fields in The User Application RTCP Packet.

Field	***Size (bit)***	***Function***

V=2	2	As in RFC1889
P	1	As in RFC1889
X	1	As in RFC1889
Subtype	5	As in RFC1889
PT	8	As in RFC1889
Length	16	As in RFC1889
TxGOVSeqNum	32	Requested GOV Sequence Number to be retransmitted
RTPPacketNum	32	Requested RTP packet number to be retransmitted, if equal to -1, retransmit the whole GOV
Spare/Unused	32	Reserved for future used

FIG.8 is an operation flowchart schematically showing the operation of a RTP/RTCP Transport Engine Server. A RTP/RTCP Transport Engine Server is responsible for segmentize and packetize GOV into RTP packets and push those RTP packets to the Client through the wireless network.

At the Client, RTP packets that contain GOV data are reassembled into a whole GOV by the help of the reference numbers in each RTP packet, i.e., TxGOVSeqNum, cr, and tr. Because UDP packet may be lost or arrive at the destination by out-of-sequence, the RTP/RTCP Transport Engine has to handle the proceeding problems. There are three types of RG (RTP GOV), i.e., New RG, Middle RG, and Last RG, whereas New RG is the first fragment of the GOV, Last RG is the last fragment of the GOV, and the rest RGs are Middle RGs. FIG. 9 is a diagram showing the structure of a complete GOV with RG (RTP GOV). There are three cases for a RTP packet arriving at the Client, i.e., the RTP packet belongs to the current expected GOV, or the GOV that should have been received before the current expected GOV, or the GOV that should be received after the current expected GOV. Correspondingly, we have three RG definitions, i.e., CurrentInSeqRG, LaggingRG, and LeadingRG. FIG. 10 shows the classification of the above three RGs.

FIG. 11 is an operation diagram showing the processing

operation of a RTP/RTCP Transport Engine Client. FIG. 12 is an operation diagram listing the main nine different processes of a RTP/RTCP Transport Engine Client upon receiving a RTP packet. In a normal transmission, a New RG should arrive at first followed by some Middle RGs, and finally a Last RG, wherein, all of the RGs should fall in the CurrentInSeqRG category, i.e., cases of 1, 4, and 7. For instance, if the Last RG of a CurrentInSeqRG is lost, the RTP/RTCP Transport Engine Client will receive a New RG without closing the current GOV. Therefore, it must close the current GOV with the incomplete Flag being set and insert the current GOV into the Data Link Buffer (Client) before it goes to handle the New RG packet. Furthermore, a special case that a GOV only has one RG packet must be handled.

The Bitrate Control Mechanism in the current invention is divided into two categories, i.e., the one to deal with the scenario that the network BW (Bandwidth) is decreasing and another one to poll the current network BW when the current streaming status is quite satisfactory so that the data bitrate could be possibly increased to match the network BW. The decision of Bitrate Control is dependent on the status of the Data Link Buffer (Client), which reflects the statistical information of Packet Loss Rate, Packet Transmission Delay, Packet Retransmission Rate, and Successful Packet Retransmission Rate.

In the current invention, the Data Link Buffer (Client) is responsible for storing GOV data, collecting the Bitrate Control Information, and issuing retransmission request. The basic attribute of the Data Link Buffer (Client) is a link of GOV nodes, which is defined as the following.

```
//this definition is for the mapping of RTP packet Num and its receiving status
//It is for the retransmission searching and lost packet writing
typedef struct RTPPktNumToRecvStatus{
        DWORD  dwRTPPktNo;                 //RTP packet's number in this GOV (the nth RTP packet)
        BOOL   bRTPPktRecvStatus;          //corresponding receiving status
} RTP_PKT_NUM_TO_RECV_STATUS;

//the definition of StreamingClientDataBufferNode structure
typedef struct StreamingClientDataBufferNode{
        LPBYTE  pGOVData;                          //the pointer that points to the GOV data buffer
        DWORD   dwGOVDataBuffSize;                 //the byte's number of the GOV data buffer
        DWORD   dwGOVDataSize;                     //the actual byte's number of GOV data
        DWORD   dwPreviousGOVBitrate;              //the bitrate of last GOV
        DWORD   dwCurrentGOVBitrate;               //the bitrate of current GOV
        DWORD   dwCurrentGOVPTS;                   //the starting PTS of current GOV
        DWORD   dwCurrentGOVDuration;              //the duration of current GOV
        DWORD   dwCurrentGOVSeqNum;                //the unique sequence number of current GOV
        DWORD   dwNumOfFrame;                      //the frame number of current GOV
        DWORD   dwCurrentGOVTransmissionSeqNum;    //the unique sequence number of current GOV
                                                   //but for transmission purpose, this value is obtained from StreamingServer

        DWORD   dwCurrGOVRTPTransSeqNum;//this value is from the RTP layer, it is for retransmission to distinguish GOV
        bool    bGOVCompleted;                     //if any RTP packet is lost for this GOV,
                                                   //this value should be false
        bool    bBlankGOV;                         //if this is a blank GOV, then is true
        bool    bRetransmitPermission;             //if the GOV is not completed and cannot be retransmitted
                                                   //this value should be false
        int     nRTPPacketSize;                    //the size of RTP packet
        int     nTotalRTPPacketNum;                //how many RTP packets for this GOV

        RTP_PKT_NUM_TO_RECV_STATUS*     m_pRTPPktRecvStatusMap;
                //this points to an array whose size is the nTotalRTPPacketNum
                //this array is to indicate the status of the RTP packets
                //true: the RTP packet with the corresponding number has been received
                // false: the RTP packet with the corresponding number has been lost
                //depending on the nTotalRTPPacketNum, this array should be dynamically adjusted

} STREAMING_CLIENT_DATA_BUFFER_NODE;
```

Moreover, there are four basic interfaces in the Data Link Buffer (Client).

(1) ReadGOV()

This is the interface that is exposed to the Decoder for reading one GOV from the Data Link Buffer (Client).

(2) InsertGOV ()

This is the interface that is exposed to the RTP/RTCP Transport Engine Client for inserting one newly received GOV.

(3) InsertBlankGOV ()

This is the interface that is exposed to the RTP/RTCP Transport Engine Client for inserting a blank GOV that has no data into the buffer.

(4) InsertGOVRTPPacket ()

This is the interface that is exposed to the

RTP/RTCP Transport Engine Client for inserting one RTP packet payload that belongs to a certain GOV into the buffer.

In the current invention, there are three basic types of GOV, wherein,

(1) Complete GOV

Refers to those GOVs whose RTP packets can be received by the RTP/RTCP Transport Engine Client in-sequence, fully, correctly and in time; therefore, it can be reconstructed by the RTP/RTCP Transport Engine Client successfully. Complete GOV is directly inserted into the Data Link Buffer (Client) as a Complete GOV node through the interface of InsertGOV(). The sum of duration of all the Complete GOVs in the Data Link Buffer (Client), which have not been read by the Decoder, is called the Remaining GOV Playback Time.

(2) Incomplete GOV

Refers to those GOVs whose RTP packets are received partially due to the Packet Loss or long transmission delay. The RTP/RTCP Transport Engine Client can only reconstruct a GOV with some data being absent and insert the Incomplete GOV into the Data Link Buffer (Client) through the interface of InsertGOV(). The system will try to recover the absent data of the Incomplete GOV later by retransmitting those lost RTP packets if possible (as long as the original GOV is still in the Data Link Buffer (Server)). If retransmission is successful and the absent data is recovered through the interface of

InsertGOVRTPPacket(), the Incomplete GOV is recovered to Complete GOV and its duration will be added into the Remaining GOV Playback Time in the next calculation.

(3) Blank GOV

Refers to those GOVs whose RTP packets are not received but whose following GOV(s) is(are) received by the RTP/RTCP Transport Engine. In order to keep the sequence of the received GOVs, the RTP/RTCP Transport Engine Client will create a Blank GOV and insert it into the Data Link Buffer (Client) through the interface of InsertBlankGOV(). A Blank GOV resides at its should-be position in the link and is recovered by retransmitting the whole original GOV later (as long as the original GOV is still in the Data Link Buffer (Server)). If retransmission is successful and the absent data is recovered through the interface of InsertGOVRTPPacket(), the Blank GOV is recovered to Complete GOV and its duration will be added into the Remaining GOV Playback Time in the next calculation.

In the current invention, the collecting of Bitrate Control information is triggered when the Decoder is taking a GOV out of the Data Link Buffer (Client) whereas the checking of retransmission tryouts is triggered when the RTP/RTCP Transport Engine Client is inserting a GOV into the Data Link Buffer (Client).

FIG. 13 is a flowchart showing the process of GOV insertion in the Data Link Buffer (Client), whereas FIG. 14 is a flowchart showing the process of GOV reading in the Data Link Buffer (Client). FIG. 15 is a time sequence diagram showing the Bitrate Control message flows in the current invention,

whereas FIG. 16 is a time sequence diagram showing the Retransmission message flows in the current invention.

FIG. 17 is a diagram showing the process of making Bitrate Control Decision. The key factor is the Total Remaining GOV Playback Time in the Data Link Buffer (Client), in other words, how long will the Decoder play back only based on the current buffered GOVs without considering any new incoming data. Among Complete GOV, Incomplete GOV, and Blank GOV, only Complete GOV (i.e., correctly recovered GOV) is considered as effective GOV when calculating the Total Remaining GOV Playback Time. The data buffer monitoring scheme, which ultimately realizes the variable bitrate control, is a Sliding Window Monitoring system. At first, the Upperbound, the Middlevalue, and the Lowerbound buffer levels are defined, which are measured by the playback time in the buffer. Those buffer levels construct the Decision Sliding Window, in which the initial playback time/current playback time is set to be the Middlevalue. When the Total Remaining GOV Playback Time falls in between the Upperbound and the Lowerbound, the network bandwidth is said to be acceptable, i.e., good enough to support the current streaming bitrate. There is no necessary to adjust the encoding bitrate for such case. If the Total Remaining GOV Playback Time is in the region of (Upperbound, Middlevalue], it means that sometimes in the buffer, the GOV leaving rate (caused by Encoder taking data) is slower than the GOV arriving rate (caused by network data arriving in), in other words, the Decoder's decoding rate is slower than the data bitrate. If the Total Remaining GOV Playback Time is equal or over the Upperbound, it means that the Decoder's decoding rate is too slow. In order to prevent the Decoder from

overload, we have to decrease the Encoder's encoding bitrate to a reasonable stage to match the Decoder's throughput regardless of the health state of the network bandwidth. On the other hand, if the Total Remaining GOV Playback Time is in the region of [Middlevalue, Lowerbound), it means that sometimes in the buffer, the GOV leaving rate (caused by Encoder taking data) is faster than the GOV arriving rate (caused by network data arriving in), in other words, packet loss happens or the transmission delay is a bit long. So retransmission is necessary. If the Total Remaining GOV Playback Time is equal or below the Lowerbound, it means that the network cannot sustain the current data bitrate any more, either because of frequent packet loss or long transmission delay. In order to prevent the Decoder from hungry of data, we have to decrease the Encoder's encoding bitrate to a reasonable stage to match the network throughput.

As the Decoder can be easily upgraded to a high performance machine to solve its throughput bottleneck, the Bitrate Control mechanism in the current invention mainly focus on the network deterioration case, which physically is out of our control. If the packet loss happens or packet transmission delay is longer than one GOV playback time, or any reason that will result in the absence of the expected GOV, the current buffer level (the Total Remaining GOV Playback Time) will drop. For instance, when a playback starts, the buffer level is set at the Middlebvalue. If packet loss happens, the buffer level will drop to (Middlevalue – 1) after the Decoder takes one GOV from the buffer as the expected GOV cannot arrive on time. If the lost packet can be retransmitted in time, the buffer level will go back

to Middlevalue. But if not, and more packet losses occur, the buffer level will continuously drop because there is no new GOV coming in time to fill up the vacancies after the Decoder takes out GOVs. If it reaches the Lowerbound, we conclude that the current (past) network bandwidth cannot support the current data bitrate. We feed back the decision to the Encoder and move the Decision Window downwards by setting the current Lowerbound as the next-to-be Middlevalue and setting the next-to-be Upperbound and Lowerbound with the predefined Steps (Distances) compared to the next-to-be Middlevalue. If the situation becomes worse, the same downward adjustment will continue. Because in live streaming system, the buffer cannot be re-filled without pausing or stopping the decoding process, it is unavoidable that the decoding process must be paused or stopped in order to re-fill the buffer when the 0 Lowerbound is reached, i.e., there is no Complete GOV in the buffer any more. For such case, the buffer decision levels will be reset to the initial values, whereby the buffer must be re-filled to the initial Middlevalue then the playback can be resumed.

The main process of Bitrate Control can be simply described as below:

(1) Collect the Current Remaining GOV Playback Time in the buffer,

(2) Compare the Current Remaining GOV Playback Time to the current Decision Sliding Window,

(3) Make decision,

(4) If need to change the encoding bitrate, then send the message to the Encoder, adjust the Decision Sliding Window, and reset the Polling timer; otherwise remember the current state and check whether the Polling

timer is triggered,

(5) If the Polling timer is triggered, start Polling,

(6) Stop Polling and reset the Polling timer,

(7) If Polling succeeds, change the encoding bitrate,

(8) Repeat from step (1).

FIG.17 is a diagram showing the process of Making Bitrate Control Decision.

FIG.18 is an illustrative diagram showing the basic definitions of the Bitrate Control mechanism. FIG.19 is an illustrative diagram showing the normal playback scenario of the Bitrate Control mechanism. FIG.20 is an illustrative diagram showing the network deterioration scenario of the Bitrate Control mechanism. FIG.21 is an illustrative diagram showing the Decoder's poor throughput scenario of the Bitrate Control mechanism.

In the current invention, the Polling process is designed to verify the availability of the next data bitrate stage. If the performance of the current network streaming is quite satisfactory and it has been lasted for a certain period, the network BW is much probably higher than the current data bitrate. The network bandwidth polling is triggered by the Polling Timer. Since during the Polling procedure the normal streaming is kept on, the Polling and the current streaming will share the same bandwidth. Therefore, The Polling bitrate is set as the difference between the next data bitrate stage and the current data bitrate. The Polling is handled by the BitrateAdapter (Client) and the BitrateAdapter (Server), whereas the BitrateAdapter (Server) pushes RTP packets to the BitrateAdapter (Client) according to the Polling bitrate. If the Polling affects the current streaming performance or the packet loss rate of the Polling exceeds a threshold, the network BW cannot support the next data bitrate stage. Otherwise, the encoding bitrate can be increased to the

next higher stage. Moreover, if the condition of the Polling is met again, the Polling process will be conducted repeatedly until the maximum encoding bitrate is reached.

The principle of Polling is that if the Remaining GOV Playback Time has been in a certain Sliding Window for a period long enough (e.g., 30 seconds), the polling should be processed to check whether the network can support higher bitrate. If before the Polling Timer is triggered, the Sliding Window is adjusted, then the Polling Timer should be reset immediately. FIG. 22 is a time sequence diagram showing the polling process.

There is another kind of polling for auto-setting the initial streaming bitrate. This is also done by the negotiation between the BitrateAdapter (Client) and the BitrateAdapter (Server). The polling will start at the middle level in the bitrate stage list. If the polling is failed, then the next lower bitrate stage will be automatically chosen for the next polling. On the other hand, if the polling is successful, the next higher bitrate stage will be automatically chosen for the next polling. These procedures will repeat until a bitrate stage is found whereby the polling on itself is successful but the polling on the next higher bitrate of it is failed. This bitrate stage then is set as the initial streaming bitrate. FIG. 23 is a diagram showing the auto-negotiation procedure.

What is claimed is:

1. An MPEG-4 Live Unicast Video Streaming System for use in wireless network including an end-to-end congestion control mechanism that can automatically and dynamically adjust the data-bitrate/transmission-bitrate according to the available network bandwidth, wherein,

(1) A Rate Adaptive MPEG-4 Simple Profile Encoder

a. Generates the MPEG-4 Simple Profile live video data,

b. Pushes live video data to the Streaming Server by HTTP/TPC through LAN,

c. Adjust the encoding bitrate in accordance with the requirement from the Streaming Server, and

(2) A Streaming Server

a. Has a Data Receiver module to receive live MPEG-4 video data by HTTP/TCP from the Rate Adaptive MPEG-4 Simple Profile Encoder through LAN,

b. Has a RTSP Server module to handle the streaming session,

c. Has a RTP/RTCP Transport Engine Server module to segmentize the MPEG-4 data on the base of GOV, packetize each GOV as the payload of RTP packets, and pushes those RTP/UDP packets to the Client through wireless network according to the bitrate of each GOV, whereas RTCP is implemented for transporting retransmission request and reply,

d. Has a Bitrate Adapter (Server) module to implement the Bitrate Adaptation protocol and Network Bandwidth Polling protocol to allow the Streaming Server to proceed bitrate control tasks with the Client, whereas the bitrate control decision is to be forwarded to the Rate Adaptive MPEG-4 Simple Profile Encoder,

e. Has a Data Link Buffer (Server) to store the MPEG-4 GOV data,

(3) A Client

1. Has a Rate Adaptive MPEG-4 Simple Profile Decoder to decode the received MPEG-4 GOV data and render the pictures,
2. Has a RTSP Client module to handle the streaming session,
3. Has a RTP/RTCP Transport Engine Client to receive the RTP/UDP packets from the Streaming Server through wireless network, depacketize the payload of RTP packets to GOV, and desegmentize GOVs to MPEG-4 video stream, whereas RTCP is implemented for transporting retransmission request and reply,

f. Has a Bitrate Adapter (Client) module to implement the Bitrate Adaptation protocol and Network Bandwidth Polling protocol to allow the Client to proceed bitrate control tasks with the Streaming Server, whereas the bitrate control decision is to be forwarded to the Streaming Server,

4. Has a Data Link Buffer (Client) to store GOV data, monitor its own status, collect the bitrate control information, and forward the information to the Bitrate Adapter (Client) module.

2. The Data Link Buffer (Server) for use in a Streaming Server according to claim1,

wherein, the MPEG-4 Simple Profile video stream data is stored in as a link of GOVs with the related information such as GOV bitrate, GOV duration, GOV size, etc, and

wherein, interfaces such as inserting a GOV, reading out a GOV, and searching a GOV are exposed for other modules, and

wherein, if the speed of GOV reading is slower than the speed of GOV inserting, overwriting the old unread GOV is

allowed with resynchronization of the Read and Write pointers by resetting the buffer status and dropping the rest unread GOVs.

3. The RTP/RTCP Transport Engine Server for use in a Streaming Server according to claim 1,

wherein, each GOV is segmentized and packetized into RTP packets, and then one RTP packet is packed as the payload of one UDP packet and is pushed to the Client through the wireless network according to the data bitrate, and

wherein, any possible retransmission request is received from the Client through UDP connection, which loads the RTCP packet contain the request information, and

wherein, upon receiving the retransmission request, the required GOV is to be searched around the Data Link Buffer (Server), and if it is found, the required data or the whole GOV is to be retransmitted to the Client using RTP packets as soon as possible, otherwise, a negative acknowledgement of Forbidden-Retransmission is to be returned to the Client through RTCP channel.

4. The Bitrate Adapter (Server) for use in a Streaming Server according to claim 1,

wherein, bitrate control information is received from the Client, and bandwidth polling is proceeded with the cooperation of the Client, and

wherein, bitrate decision is forwarded to a Rate Adaptive MPEG-4 Simple Profile Encoder through TCP connection.

5. The Data Link Buffer (Client) for use in a Client according to claim 1,

wherein, the MPEG-4 Simple Profile video stream data is stored in as a link of GOVs with the related information such as GOV bitrate, GOV duration, GOV size, etc, and

wherein, interfaces such as inserting a GOV, inserting a Blank GOV, inserting data of an incomplete GOV, reading out

a GOV, and searching a GOV are exposed for other modules, and

wherein, if the speed of GOV reading is slower than the speed of GOV inserting, overwriting the old unread GOV is allowed with resynchronization of the Read and Write pointers by resetting the buffer status and dropping the rest unread GOVs, and

wherein, Incomplete GOV is verified and retransmission request is sent to a RTP/RTCP Transport Engine Client, and if retransmitted data can be inserted in by the RTP/RTCP Transport Engine Client in time, Incomplete GOV is recovered to be Complete GOV, and

wherein, current buffer status is collected and sent to a Bitrate Adapter (Client) according to the Bitrate Adaptation protocol.

6. The RTP/RTCP Transport Engine Client for use in a Streaming Server according to claim 1,

wherein, RTP packets are received by UDP connection through wireless network and then desegmentized and depacketized into GOV, and

wherein, upon packet loss or packet out-of-sequence, Incomplete GOV or Blank GOV is inserted to the Data Link Buffer (Client),

wherein, any possible retransmission request is received from the Data Link Buffer (Client), and then is forwarded to the RTP/RTCP Transport Engine Server through UDP connection, which loads the RTCP packet contain the request information, and

wherein, upon receiving the retransmitted data, the specified GOV is to be searched around the Data Link Buffer (Client), and if it is found, the retransmitted data or the whole GOV is to be inserted to its position in the link, and

wherein, if Forbidden-Retransmission RTCP packet is received, the Forbidden-Retransmission flag of the specified GOV in the Data Link Buffer (Client) is to be set to forbid further retransmission request, and

7. The Bitrate Adapter (Client) for use in a Client according to claim 1,

wherein, bitrate control information is received from the Data Link Buffer (Client), and bitrate decision is made and forwarded to a Bitrate Adapter (Server) in a Streaming Server through TCP connection, and

wherein, according to the Network Bandwidth Polling protocol, a polling process is activated to work with a Bitrate Adapter (Server) in a Streaming Server, and

wherein, an auto-negotiation on the initial streaming bitrate between a Streaming Server and a Client is initiated to work with a Bitrate Adapter (Server) in the Streaming Server by using the Network Bandwidth Polling protocol.

8. The extended RTP packet structure for use in a RTP/RTCP Transport Engine Server according to claim 3, and in a RTP/RTCP Transport Engine Client according to claim 6,

wherein, additional fields are defined for depacketization and desegmentation.

9. The user application RTCP structure for use in a RTP/RTCP Transport Engine Server according to claim 3, and in a RTP/RTCP Transport Engine Client according to claim 6,

wherein, additional fields are defined for retransmission.

10. The GOV node structure for use in a Data Link Buffer (Server) according to claim 2, and in a Data Link Buffer (Client) according to claim 5

wherein, one GOV is stored in one GOV node with some related information.

11. The retransmission mechanism for use in a Data Link Buffer (Client) according to claim 5, in a RTP/RTCP Transport

Engine Client according to claim 6, and in a RTP/RTCP Transport Engine Server according to claim 3,

12. The Network Bandwidth Polling protocol for use in a Bitrate Adapter (Server) according to claim 4 and a Bitrate Adapter (Client) according to claim 7.

13. The Bitrate Adaptation protocol for use in a Data Link Buffer (Client) according to claim 5, a Bitrate Adapter (Server) according to claim 4 and a Bitrate Adapter (Client) according to claim 7.

14. The Bitrate Decision Rule with implementation of the Decision Sliding Window for use in a Bitrate Adaptation Protocol according to claim 13.

162162

ABSTRACT OF THE DISCLOSURE

An MPEG-4 LIVE UNICAST VIDEO STREAMING SYSTEM IN WIRELESS NETWORK WITH END-TO-END BITRATE-BASED CONGESTION CONTROL

With end-to-end congestion control, in an MPEG-4 Live Unicast Video Streaming System in wireless network, a Streaming Server provides real-time video-streaming to a Client by using RTP/UDP protocol. RTP/RTCP Transport Engines handle the segmentization / desegmentization and packetization / depacketizaiton of the data as well as the transmission/retransmission of the packets. A Bitrate Adaptation protocol and a Network Bandwidth Polling protocol can automatically and dynamically adjust the data-bitrate/transmission-bitrate according to the available network bandwidth; so that the continuous live video-streaming service is promised.

G00002

163163

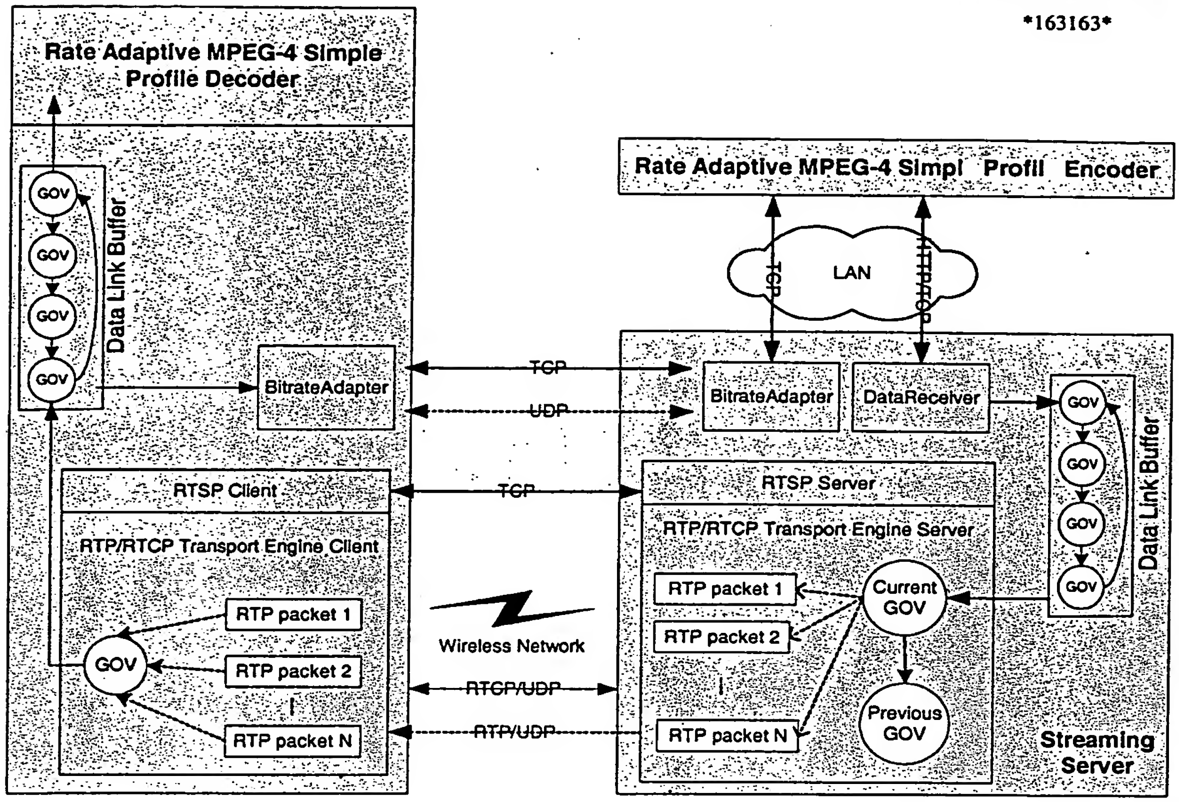

FIG. 1 Diagram showing the system architecture of the present invention.

G00002

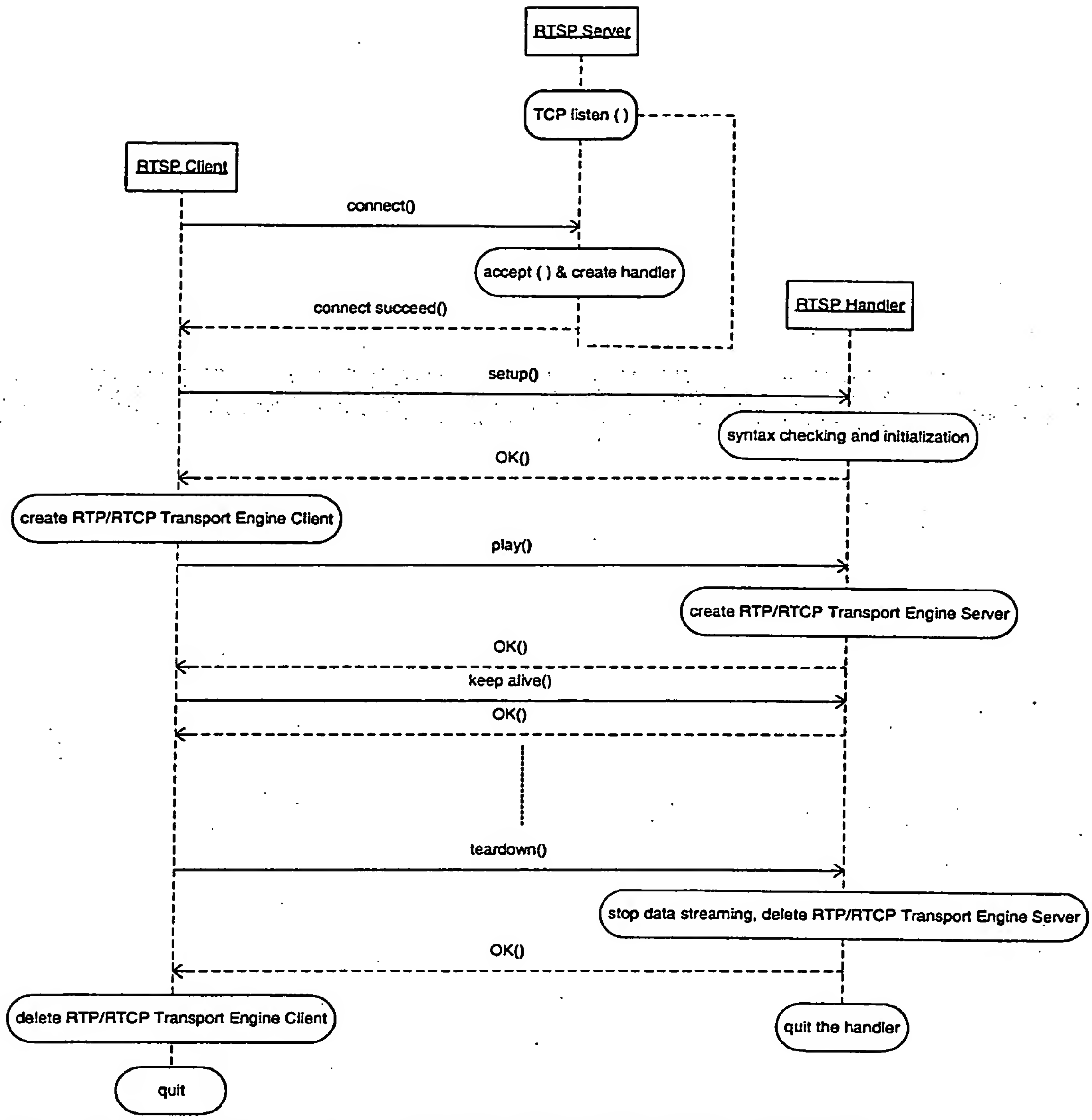

FIG. 2 Diagram showing the overview of the RTSP session procedure.

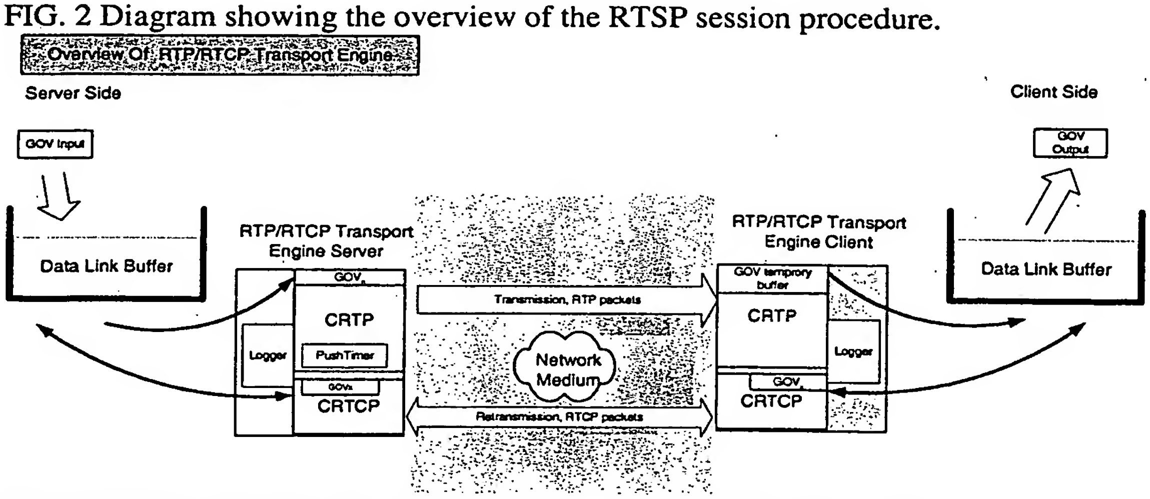

FIG. 3 Diagram showing the overview of the RTP/RTCP Transport Engine.

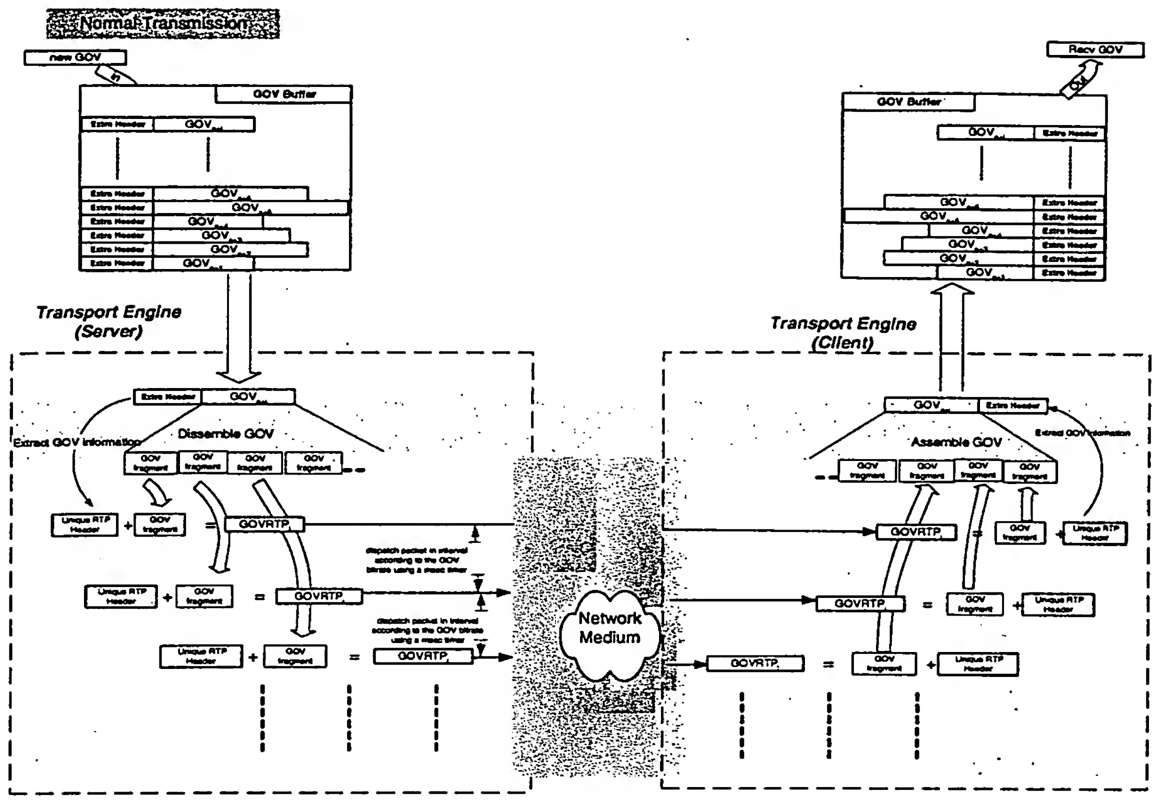

FIG. 4 Diagram showing the overview of the normal data transmission between the RTP/RTCP Transport Engines.

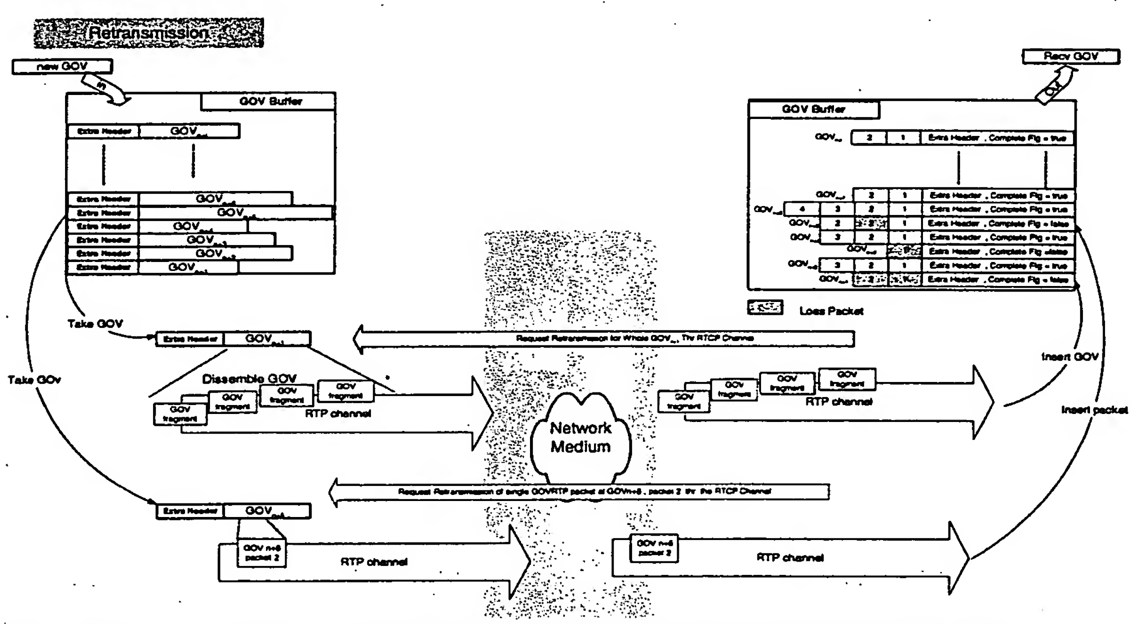

FIG. 5 Diagram showing the overview of the data retransmission between the RTP/RTCP Transport Engines.

Extended RTP

32 bits

<table>
<tr><td>V=2</td><td>P</td><td>X</td><td>CC</td><td>M</td><td>PT</td><td colspan="2">Seq Num</td></tr>
<tr><td colspan="8">TimeStamp</td></tr>
<tr><td colspan="8">SSRC</td></tr>
<tr><td colspan="7">define by profile</td><td>length</td></tr>
<tr><td colspan="8">TxGOVSeqNum</td></tr>
<tr><td colspan="6">tr</td><td colspan="2">cr</td></tr>
<tr><td colspan="8">SENTT</td></tr>
<tr><td colspan="8">dwGOVDataBuffSize</td></tr>
<tr><td colspan="8">dwGOVDataSize</td></tr>
<tr><td colspan="8">dwPreviousGOVBitrate</td></tr>
<tr><td colspan="8">dwCurrentGOVBitrate</td></tr>
<tr><td colspan="8">dwCurrentGOVPTS</td></tr>
<tr><td colspan="8">dwCurrentGOVDuration</td></tr>
<tr><td colspan="8">dwCurrentGOVSeqNum</td></tr>
<tr><td colspan="8">dwNumOfFrame</td></tr>
<tr><td colspan="8">dwCurrentGOVTransmissionSeqNum</td></tr>
<tr><td colspan="8">GOV data</td></tr>
</table>

FIG. 6 Diagram showing the structure of the Extended RTP Packet.

User Application RTCP

32 bits

<table>
<tr><td>V=2</td><td>P</td><td>X</td><td>subtype</td><td>PT</td><td>Length</td></tr>
<tr><td colspan="6">SSRC/CSRC</td></tr>
<tr><td colspan="6">TxGOVSeqNum</td></tr>
<tr><td colspan="6">RTPacketNum</td></tr>
<tr><td colspan="6">Spare/Unused</td></tr>
</table>

FIG. 7 Diagram showing the structure of the User Application RTCP Packet.

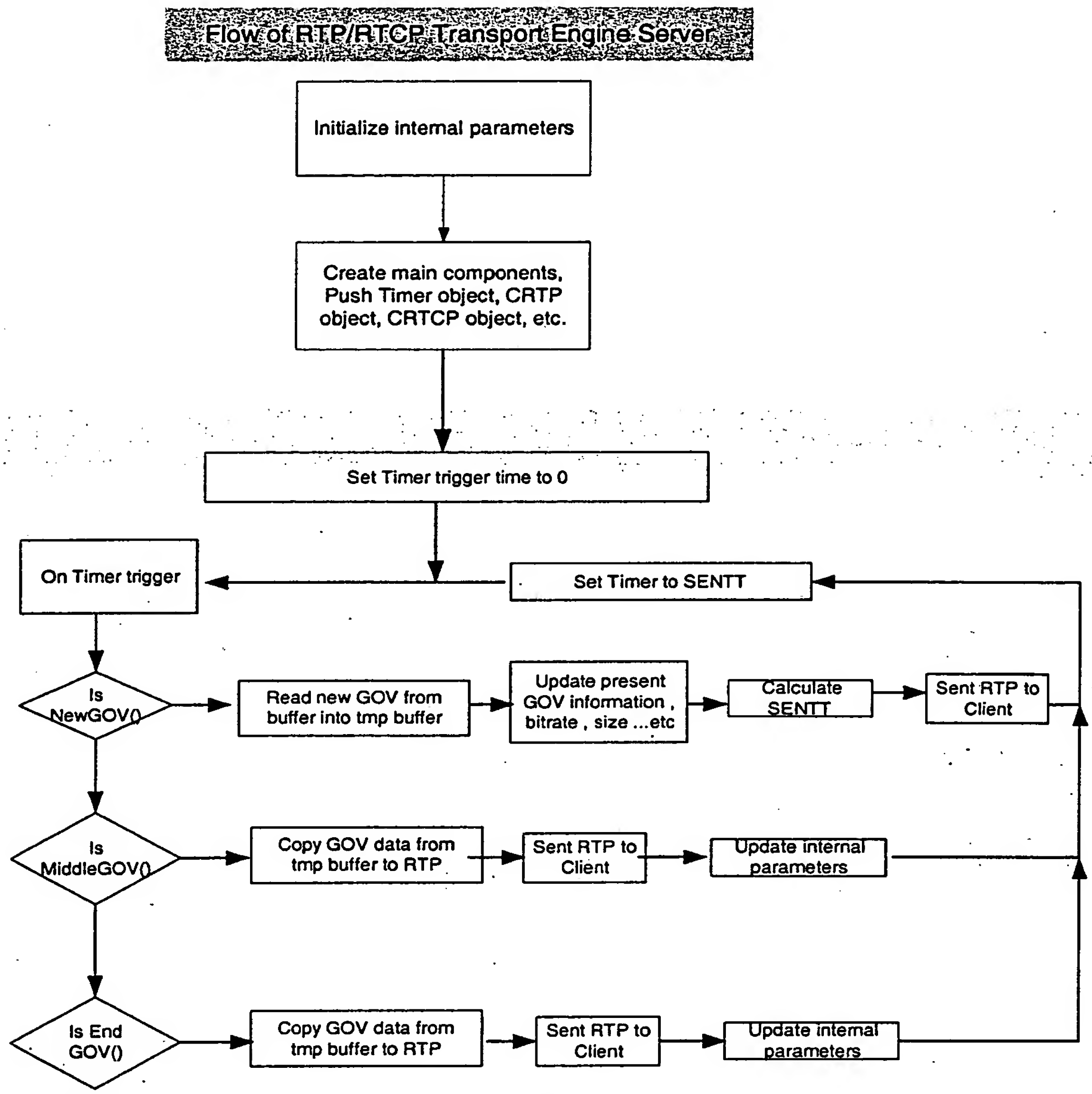

FIG. 8 An operation flowchart schematically showing the processing operation of a RTP/RTCP Transport Engine Server.

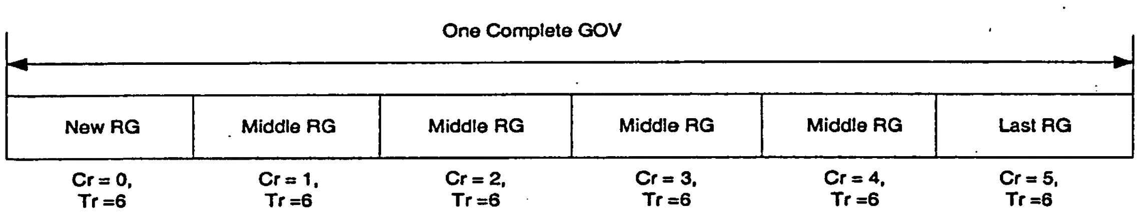

FIG. 9 A diagram showing the structure of a complete GOV with RG (RTP GOV).

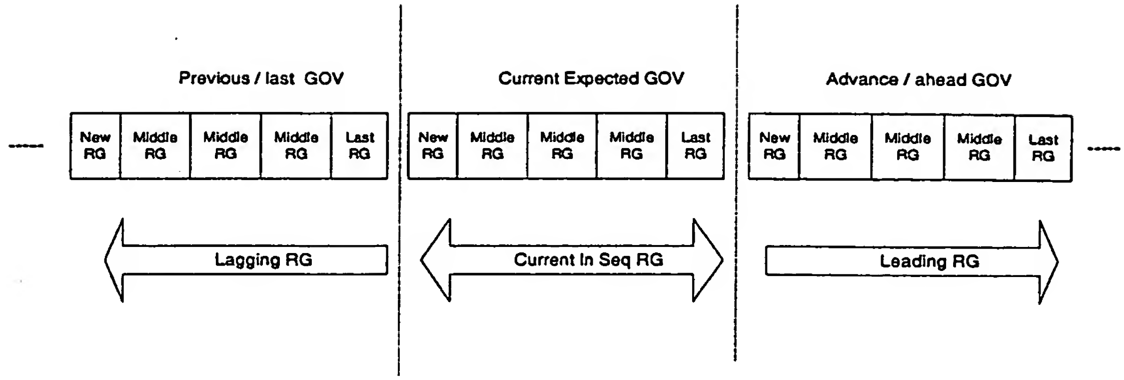

FIG. 10 A diagram showing the classification of three different RGs regarding the UDP out-of-sequence problem.

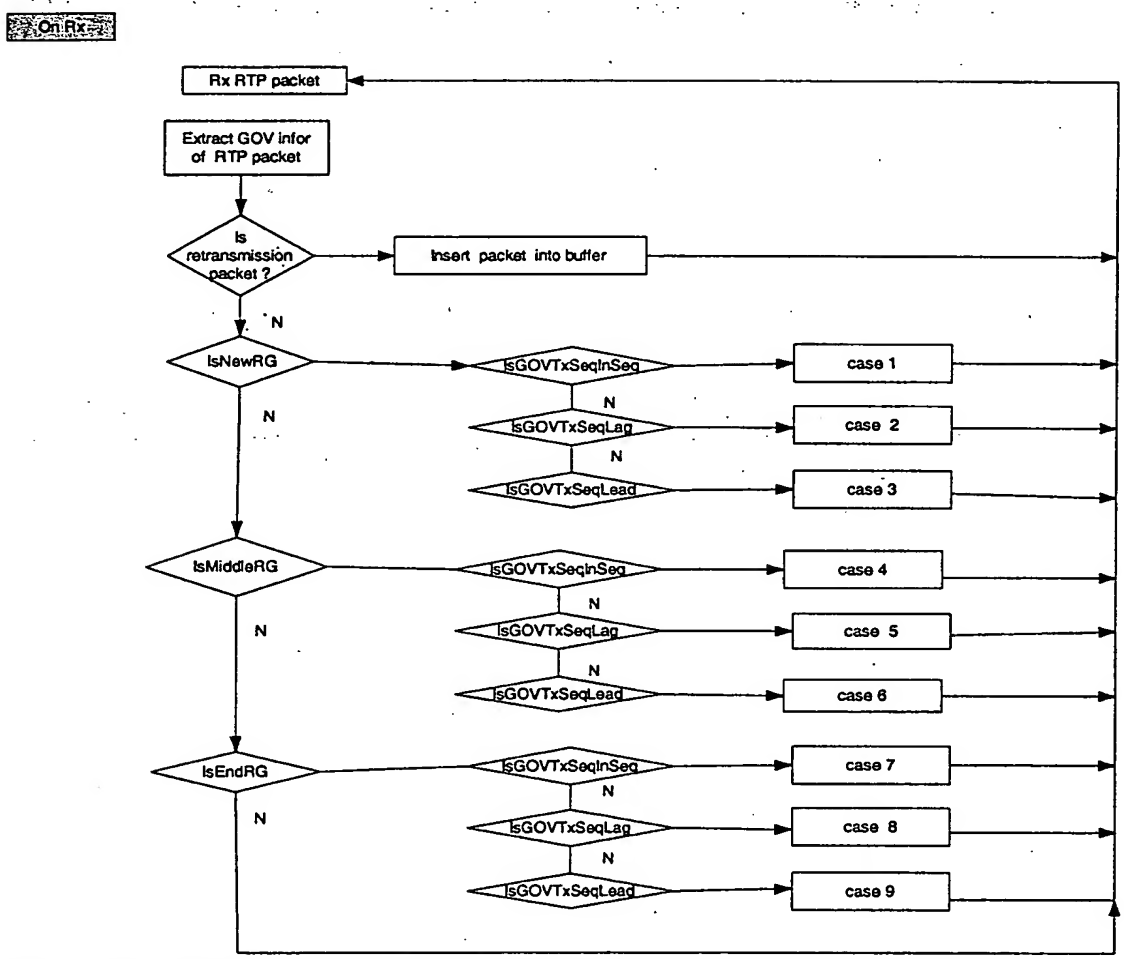

FIG. 11 An operation diagram showing the processing operation of a RTP/RTCP Transport Engine Client.

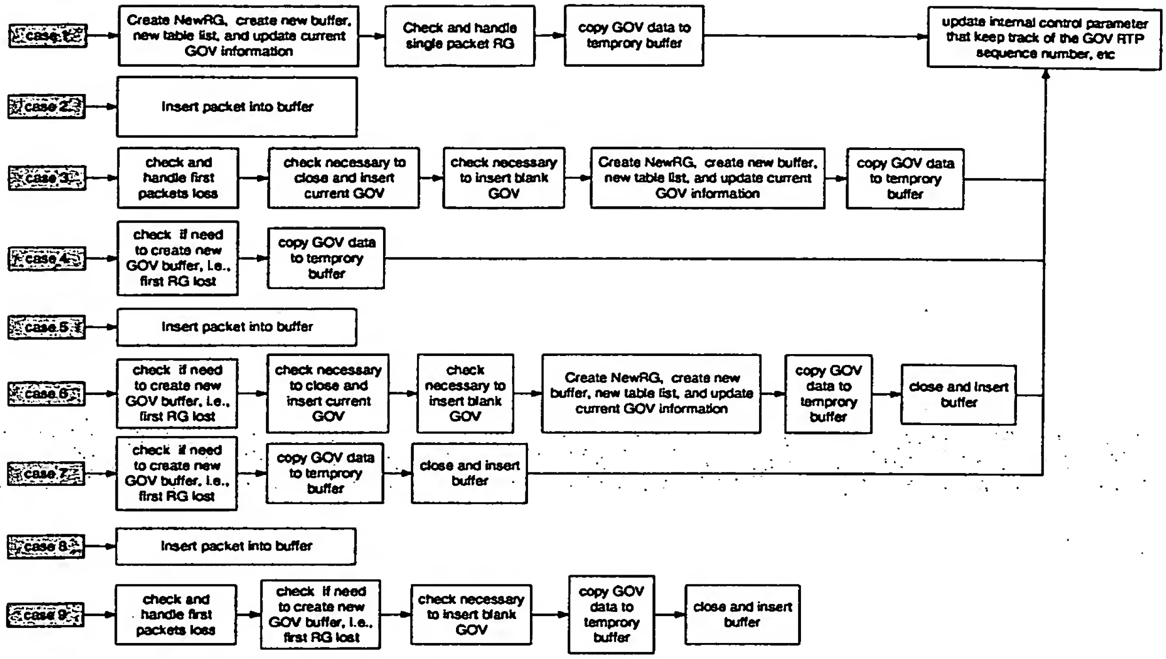

FIG. 12 An operation diagram listing the main nine different processes of a RTP/RTCP Transport Engine Client upon receiving a RTP packet.

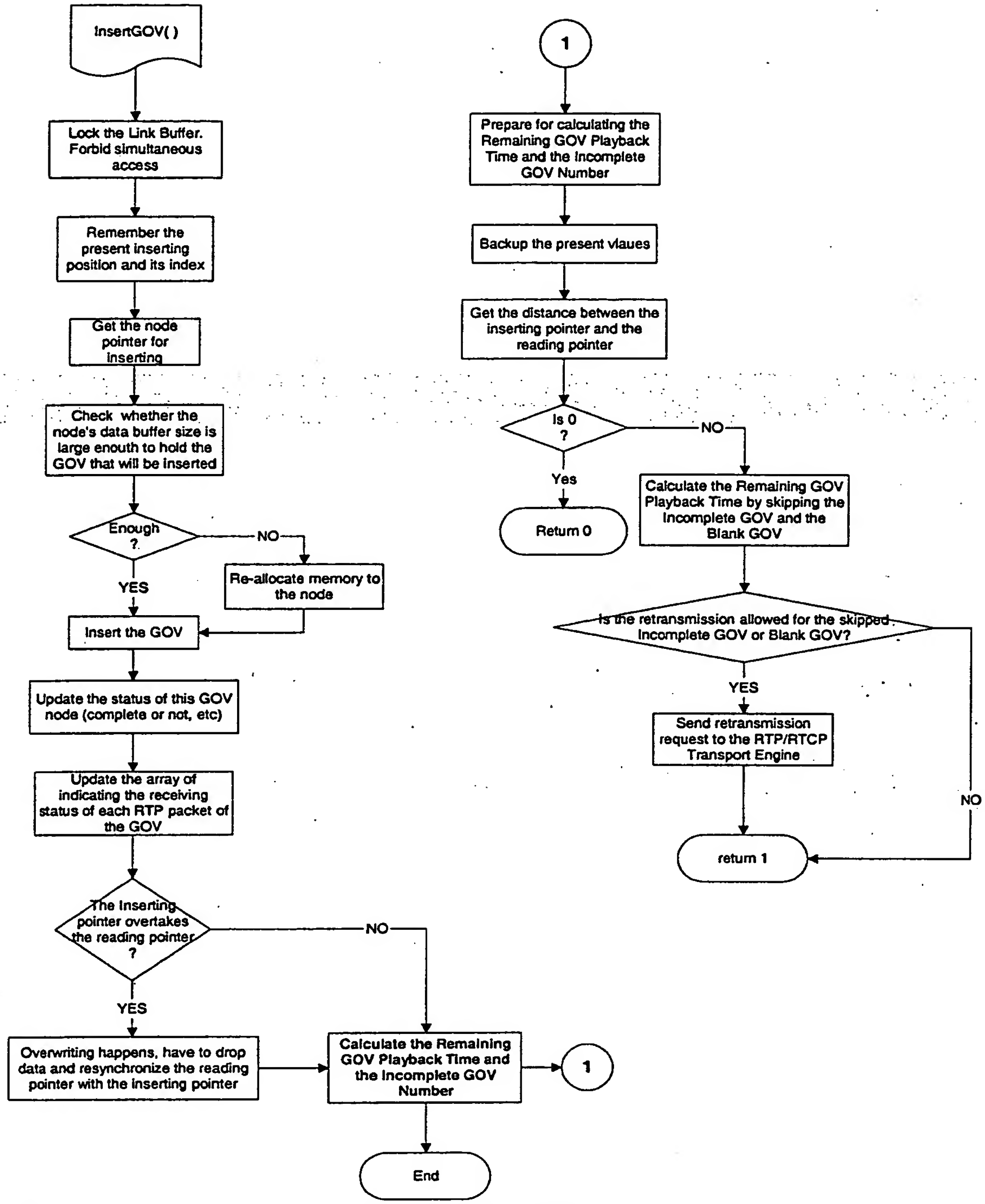

FIG. 13 A flowchart showing the process of GOV insertion in the Data Link Buffer (Client).

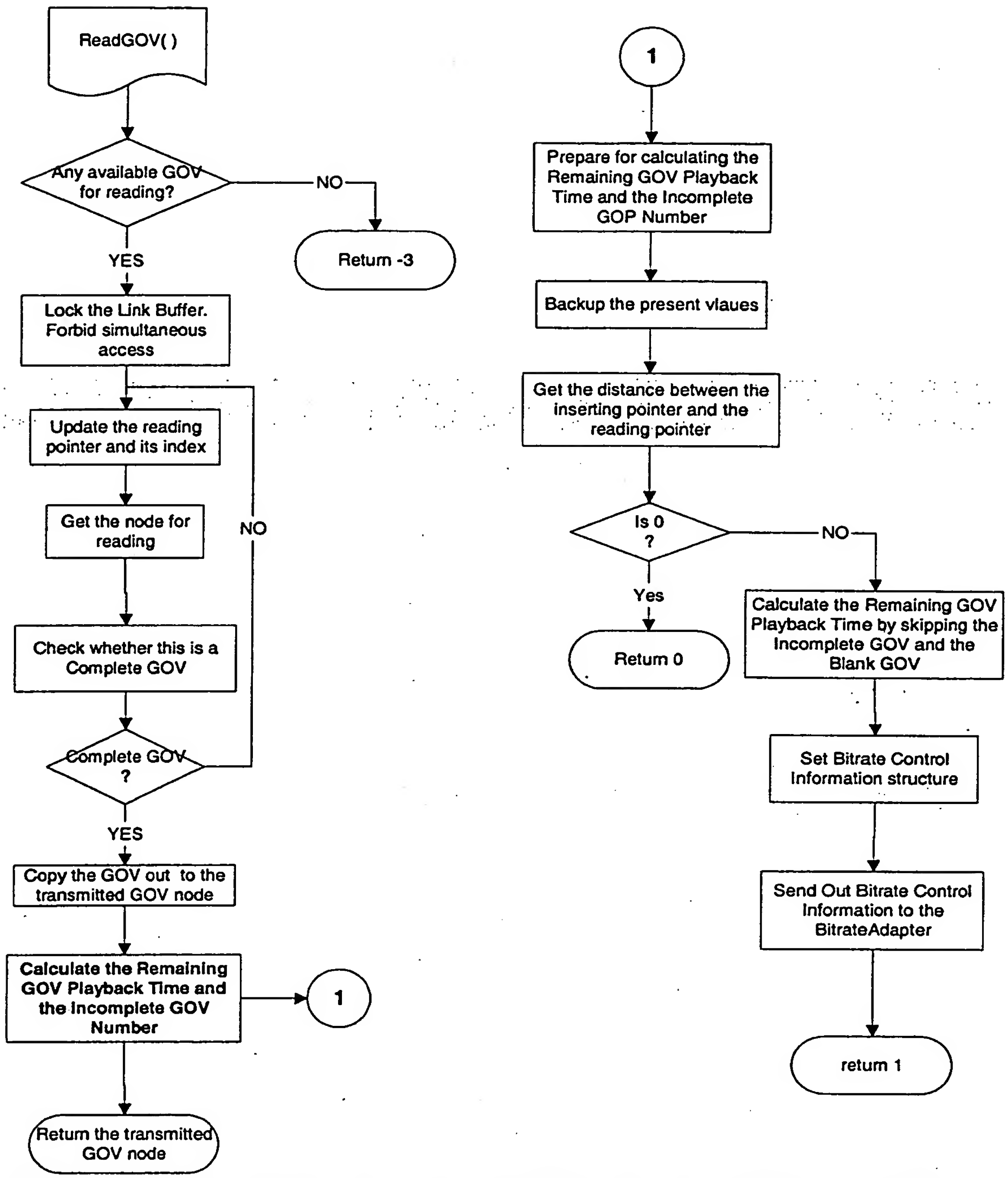

FIG. 14 A flowchart showing the process of GOV reading in the Data Link Buffer (Client).

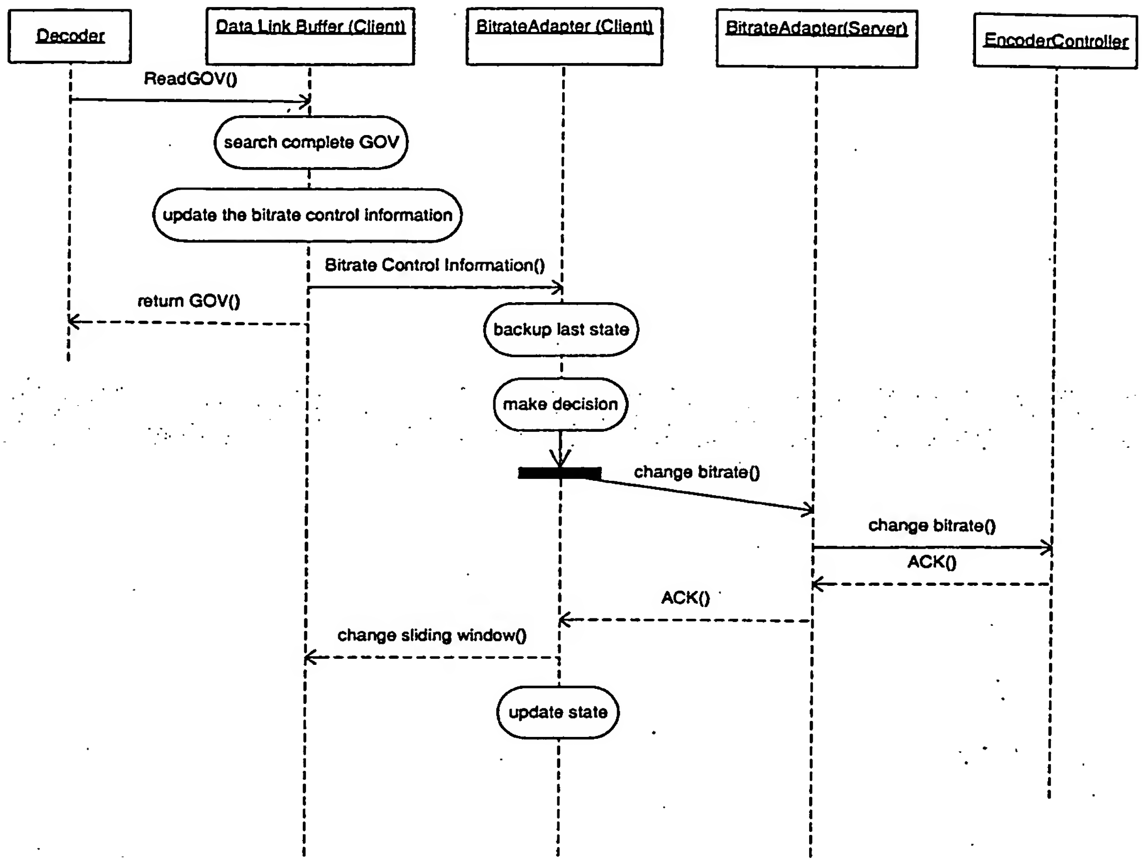

FIG. 15 A time sequence diagram showing the Bitrate Control message flows in the current invention.

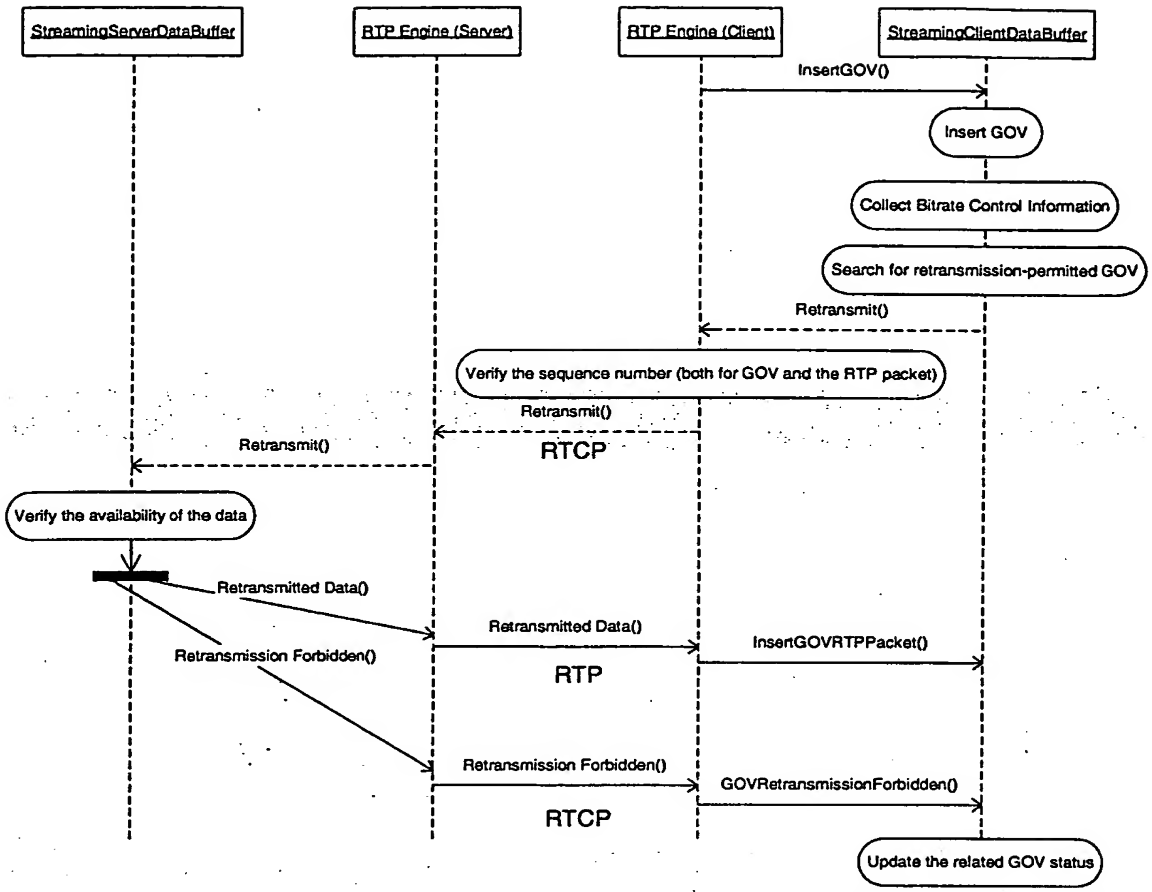

FIG. 16 A time sequence diagram showing the Retransmission message flows in the current invention.

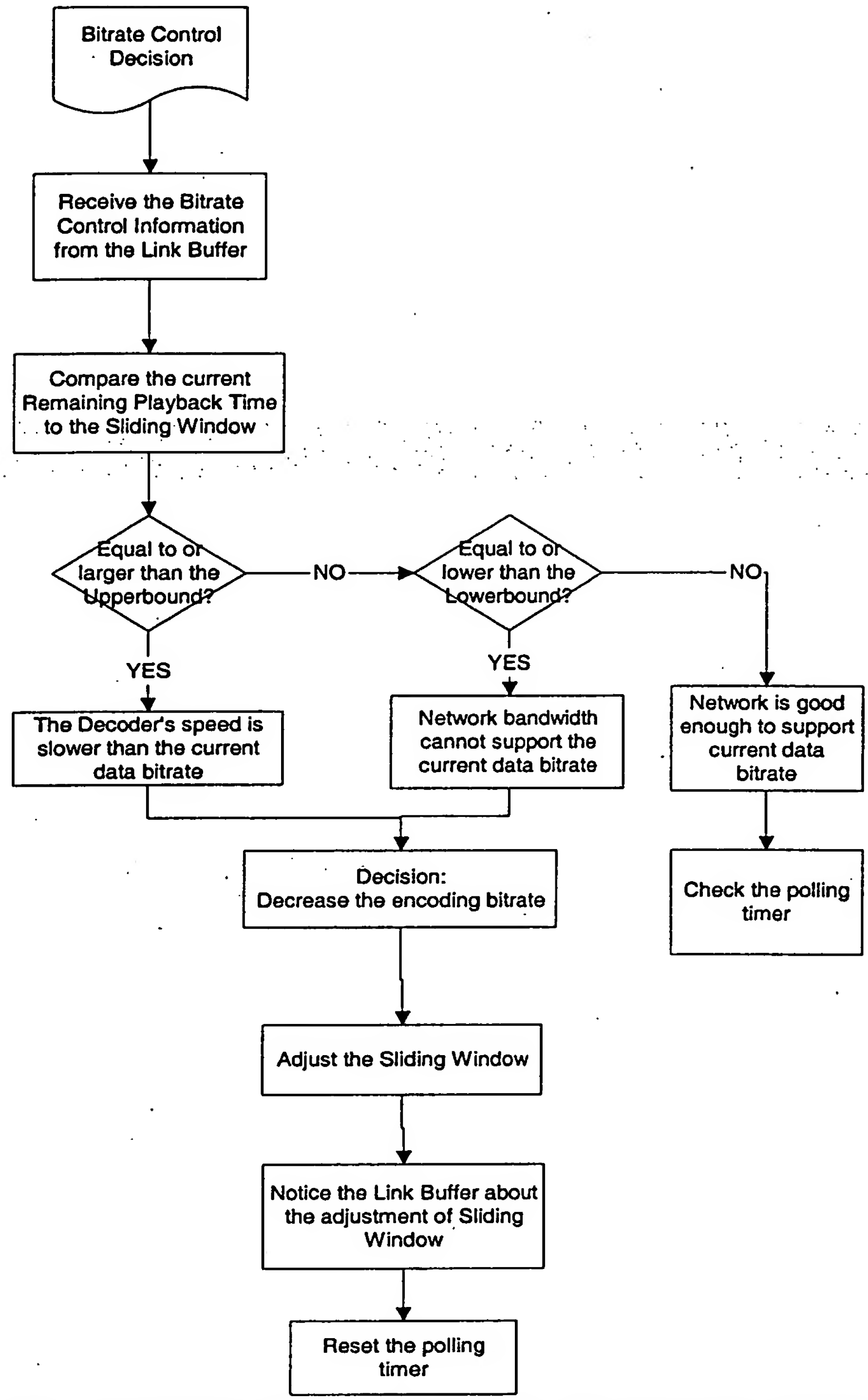

FIG. 17 A diagram showing the process of Bitrate Control Decision making.

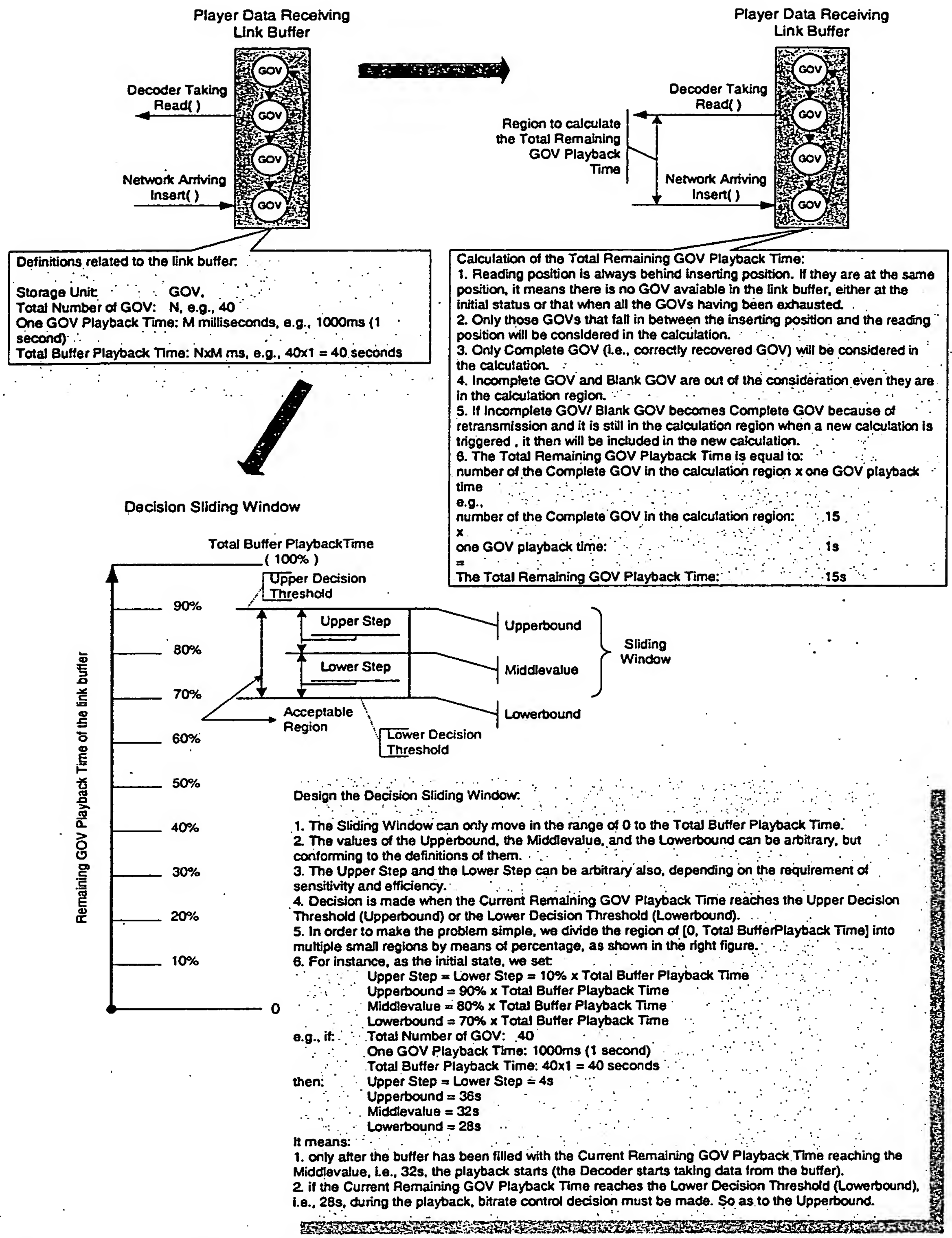

FIG. 18 An illustrative diagram showing the basic definitions of the Bitrate Control mechanism.

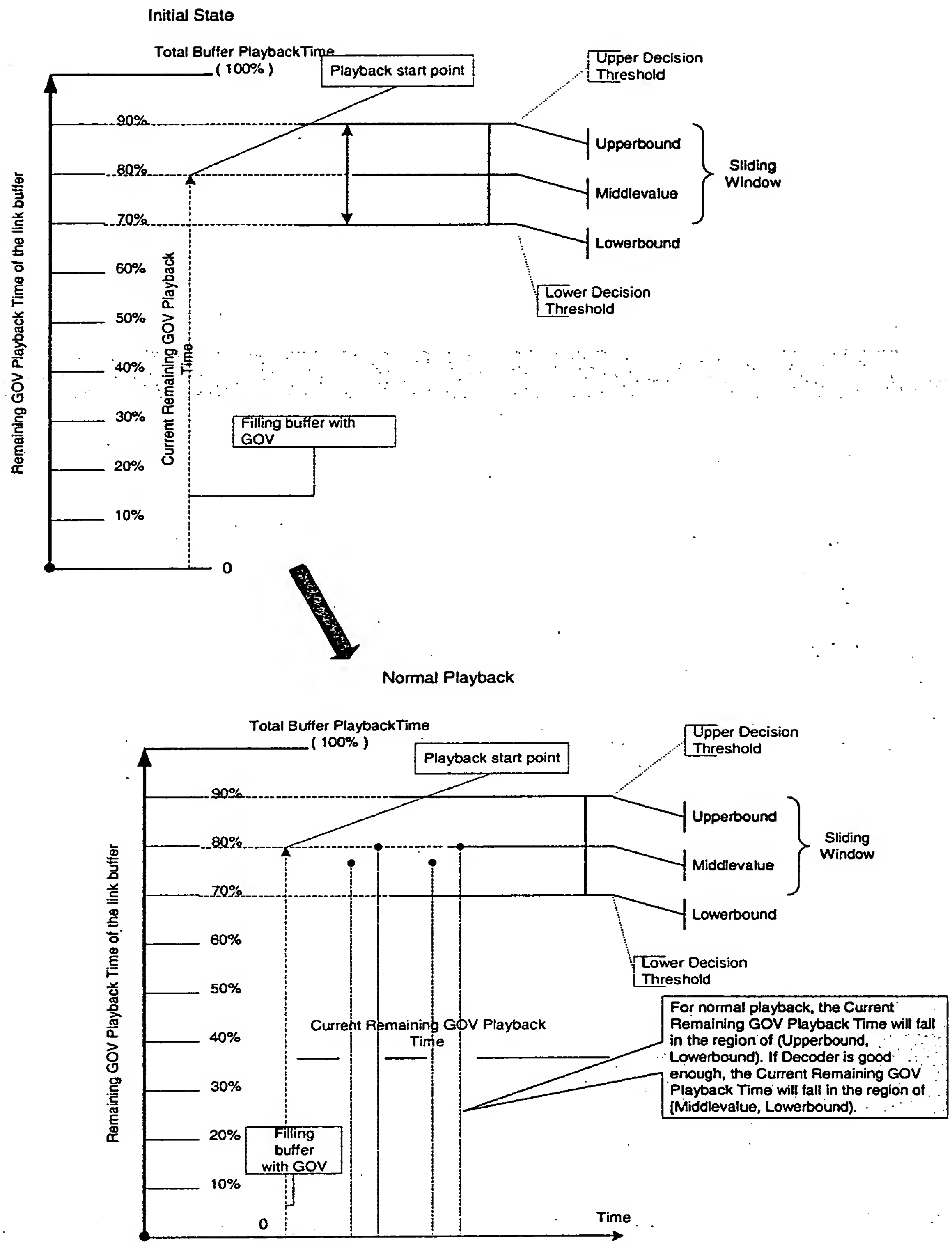

FIG. 19 An illustrative diagram showing the normal playback scenario of the Bitrate Control mechanism.

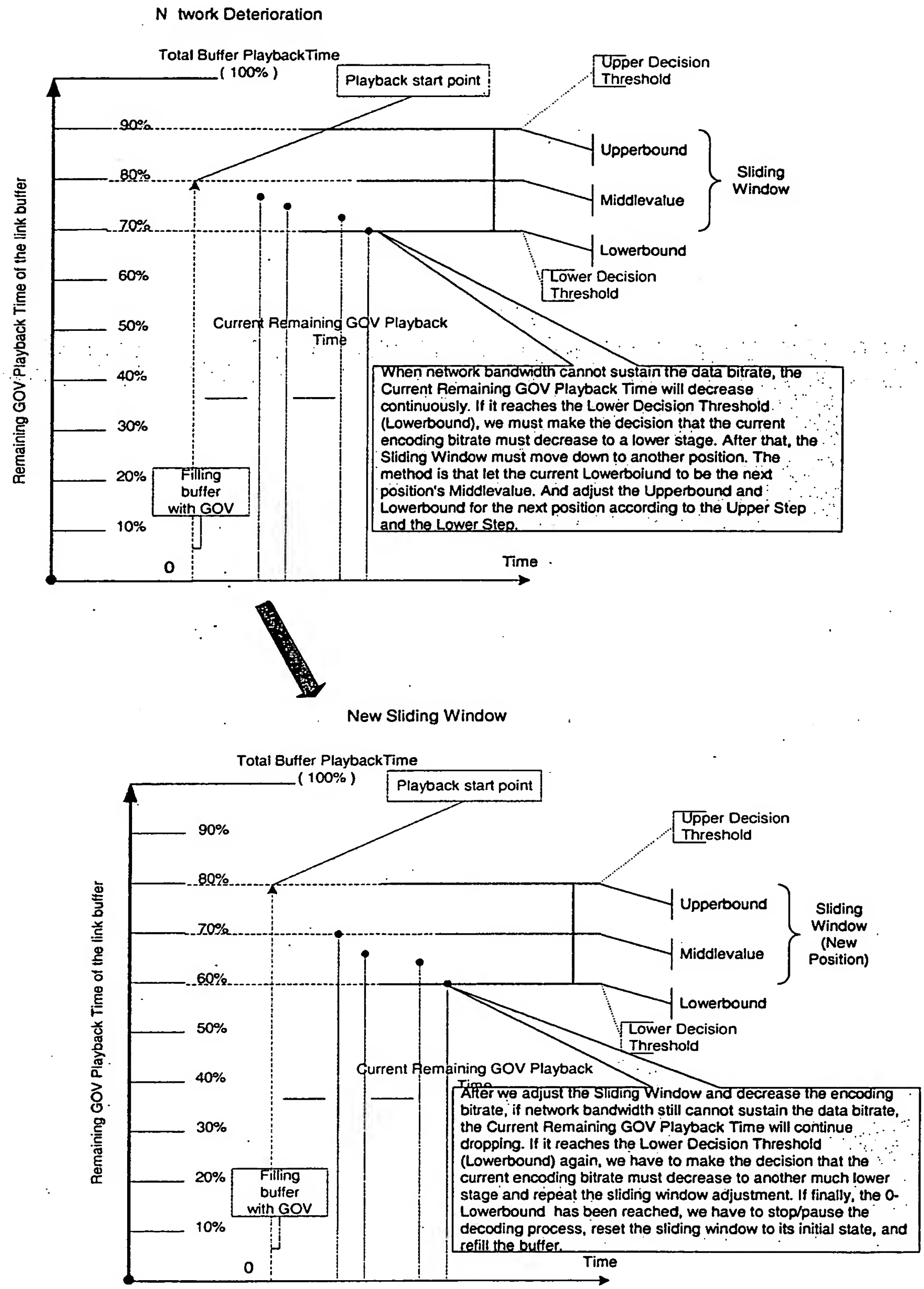

FIG. 20 An illustrative diagram showing the network deterioration scenario of the Bitrate Control mechanism.

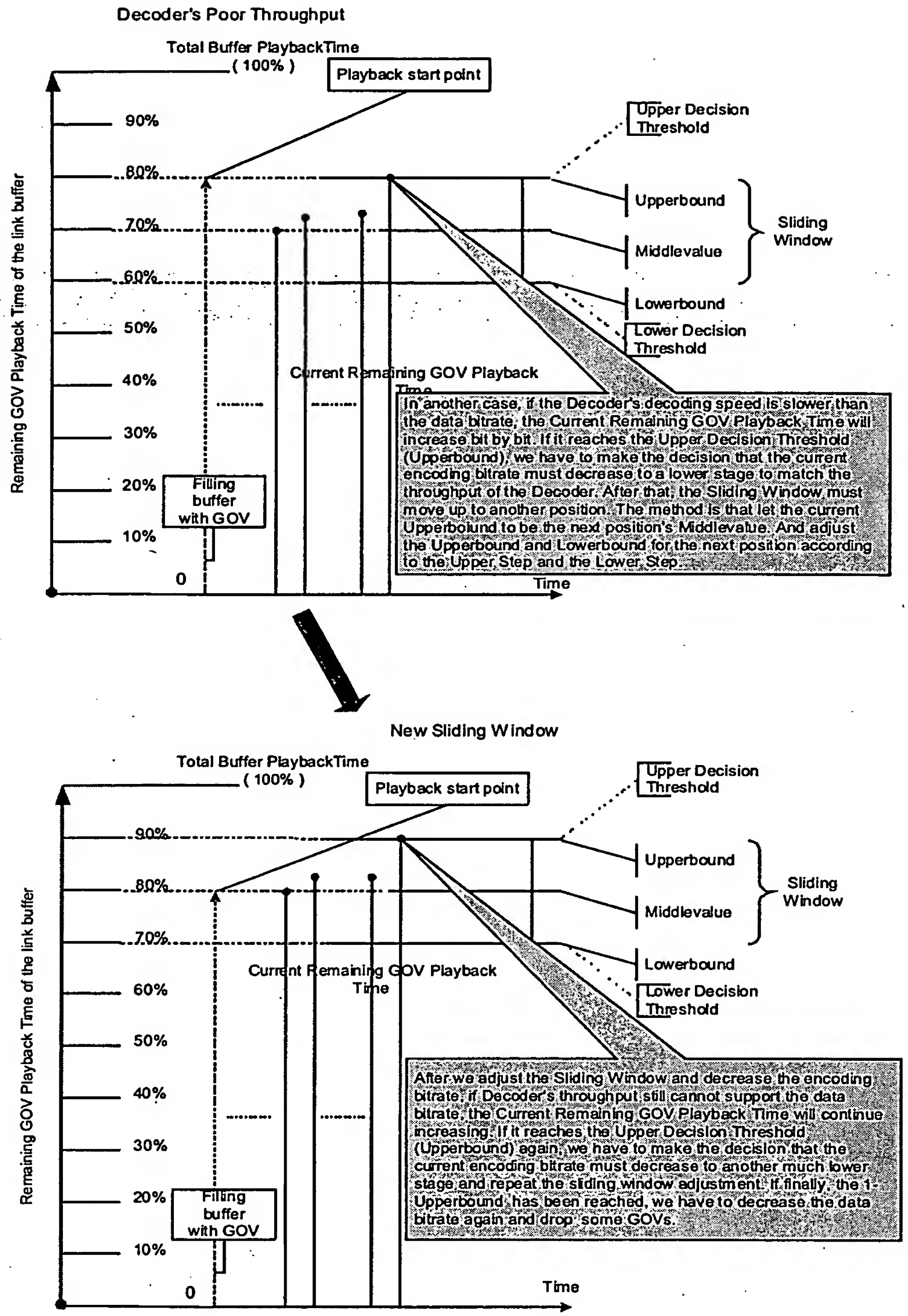

FIG. 21 An illustrative diagram showing the Decoder's poor throughput scenario of the Bitrate Control mechanism.

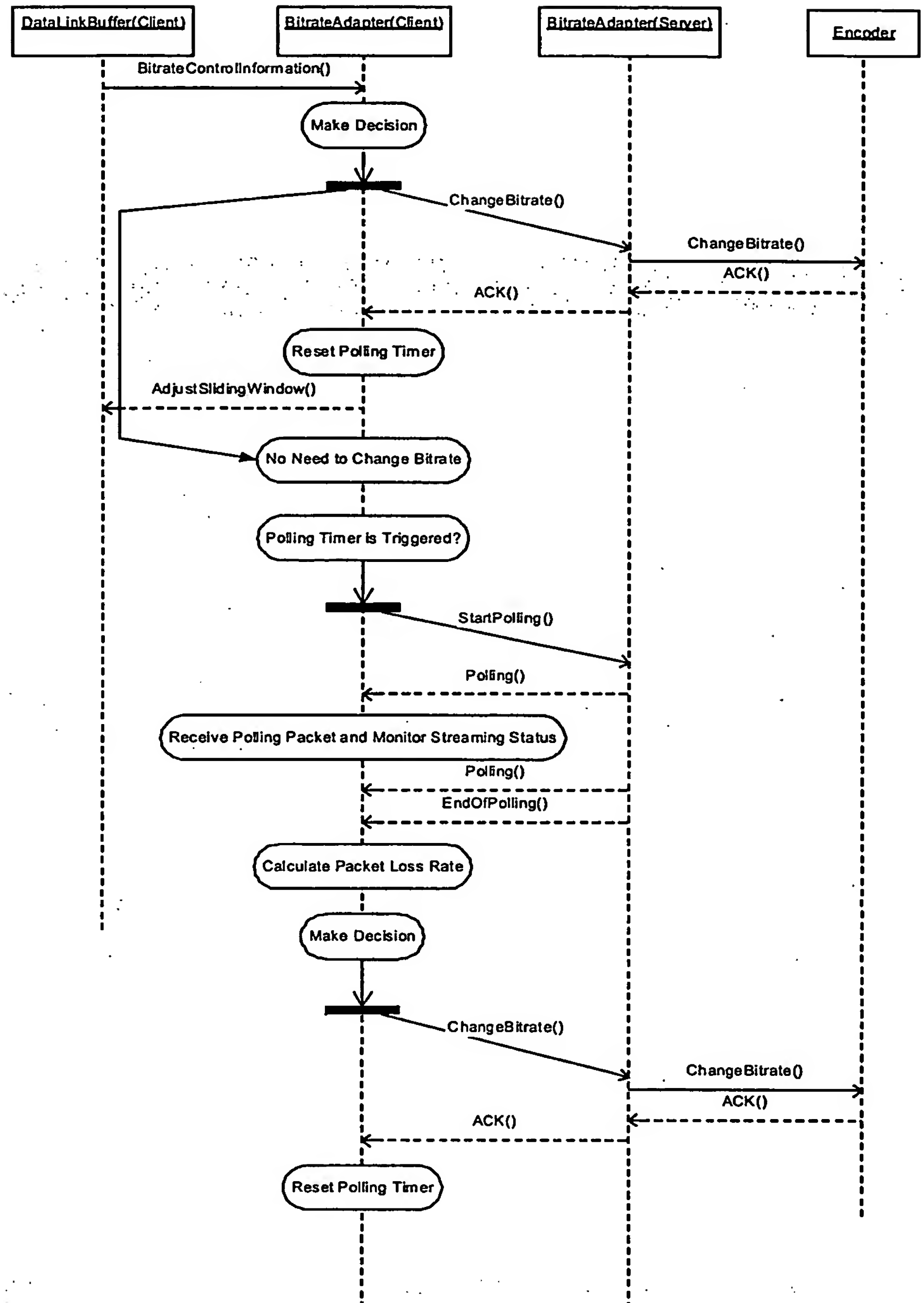

FIG. 22 A time sequence diagram showing the polling process.

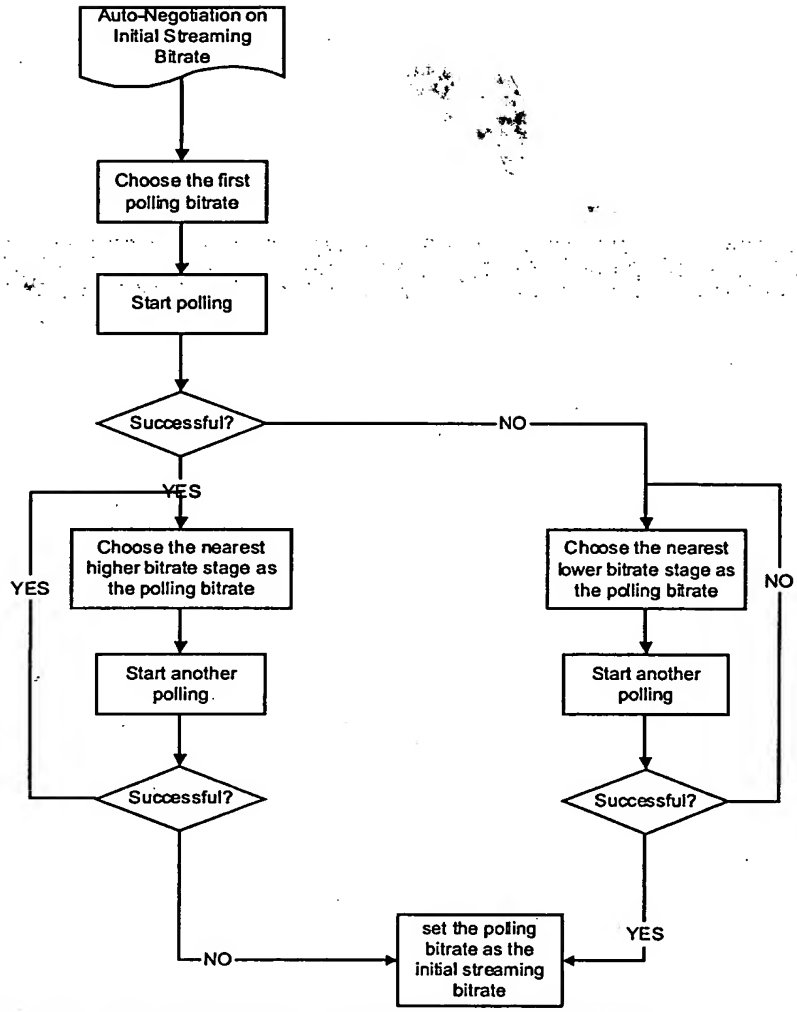

FIG. 23 A diagram showing the auto-negotiation procedure.